

Solving a bi-objective mathematical model for location-routing problem with time windows in multi-echelon reverse logistics using metaheuristic procedure

V. R. Ghezavati¹ · M. Beigi¹

Received: 7 November 2015 / Accepted: 7 June 2016 / Published online: 25 June 2016
© The Author(s) 2016. This article is published with open access at Springerlink.com

Abstract During the last decade, the stringent pressures from environmental and social requirements have spurred an interest in designing a reverse logistics (RL) network. The success of a logistics system may depend on the decisions of the facilities locations and vehicle routings. The location-routing problem (LRP) simultaneously locates the facilities and designs the travel routes for vehicles among established facilities and existing demand points. In this paper, the location-routing problem with time window (LRPTW) and homogeneous fleet type and designing a multi-echelon, and capacitated reverse logistics network, are considered which may arise in many real-life situations in logistics management. Our proposed RL network consists of hybrid collection/inspection centers, recovery centers and disposal centers. Here, we present a new bi-objective mathematical programming (BOMP) for LRPTW in reverse logistic. Since this type of problem is NP-hard, the non-dominated sorting genetic algorithm II (NSGA-II) is proposed to obtain the Pareto frontier for the given problem. Several numerical examples are presented to illustrate the effectiveness of the proposed model and algorithm. Also, the present work is an effort to effectively implement the ϵ -constraint method in GAMS software for producing the Pareto-optimal solutions in a BOMP. The results of the proposed algorithm have been compared with the ϵ -constraint method. The computational results show that the ϵ -constraint method is able to solve small-size instances to optimality within reasonable computing times,

and for medium-to-large-sized problems, the proposed NSGA-II works better than the ϵ -constraint.

Keywords Reverse logistics network · Location-routing problem · Time window · Bi-objective model · ϵ -Constraint method · NSGA-II

Introduction

During the last decade, growing attention has been paid to reverse logistics network design (RLND), which focuses on the backward network. RLN is utilized to pick up or collection, transportation and recycling of used products or end-of-life (EOL) goods by the consumers, such as electronic goods recycling (Hyunsoo et al. 2009), hazardous waste products recycling (Samanlioglu 2013), empty and aluminum soft-drink bottles recycling (Privé et al. 2006), paper recycling (Patia et al. 2008).

RLND generally refers to activities such as collection, inspection/separation, recovery, repair, recycling, remanufacturing or re-processing, disposal and re-distribution of the used products. Various researchers classified the reverse logistic process differently. Many logistic networks aim to decide on issues such as (1) locations for depots (2) allocation of customers to each established facilities, and (3) transportation networks connecting customers to facilities by vehicle routing. (4) Inventory management of goods on facilities. Nowadays, the combined two or more problem has been considered. The problem, which deals with combines the facility location problem (FLP) and the vehicle routing problem (VRP) decisions, is known in operations research context as the location-routing problem (LRP). According to Vidović et al. (2016) and Prodhon and Prins (2014), the LRPs can

✉ V. R. Ghezavati
v_ghezavati@azad.ac.ir

¹ School of Industrial Engineering, Islamic Azad University, South Tehran Branch, Tehran, Iran

be classified based on the different aspects, such as single or multiple echelons, hierarchical structure, number of facilities, number and types of vehicles, homo-/heterogeneous fleet, (un) limited/(un)capacitated fleet, facility capacity, type of input data, nature of demand, planning horizon, time windows, number of objective functions, route structure, solution space, and solution method. In this paper, we have utilized a bi-objective capacitated location-routing problem with soft time window (LRPTW) and with four layers (e.g., collection and inspection centers, recycling centers, disposal centers and customers) in reverse logistics and heterogeneous vehicle fleet with capacity. Also, the turn of the customer is considered that it has not been seen, previously, in the literature. A predefined percent of demand from each customer is assumed as returned products, and a predefined value is determined as an average scrap fraction. The model determines which depots should be opened (or established) in all echelons and identifies the collection routes from the collection centers to the customers with considering the turn of customer in first echelon.

LRP deals with determining the location of facilities and the routes of the vehicles for serving the customers under some constraints, such as facility and vehicle capacities, route time, to minimize total cost including transportation costs, vehicle fixed costs, facility location fixed costs, recycling centers operating costs and penalty cost and to satisfy demands of all customers by minimizing total time as second objective function. Notably, these two objective functions are in conflict with each other. This means that an increase in one objective leads to a decrease in another one; therefore, optimizing the network involves a trade-off between these two objectives. Furthermore, a complete sensitivity analysis is presented to investigate this model from different perspectives.

So, the main contributions of this paper that differentiate our efforts from those already published on this issue can be summarized as follows: (1) We introduce a new formulation of the CLRPTW, in which vehicle routing is considered in first echelon and facilities location is considered for collection and inspection centers, recycling centers. (2) We support both collection and inspection processes in one facility for reducing cost. (3) Allow to trade off between two important objectives in this area, i.e., the total costs and the total network responsiveness by reducing maximum traveling time to offer different compromise efficient solutions to the decision makers. (4) We consider turn of the customer, the soft time window with penalty cost and the load of vehicles after leaving every node of customer in our LRP. (5) We propose exact solution in GAMS for solving small-size instances and meta-heuristic solution methods, non-dominated sorting genetic algorithm II (NSGA-II), based on the new formulation,

providing the means to solve large-size instances, and to compute tight gaps for small instances.

The remainder of this paper is organized as follows. The brief surveys' literature on the CLRP and related problems are defined in "[Literature review](#)". The problem definition and proposed bi-objective mathematical formulation are elaborated in "[Problem description and mathematical formulation](#)", and the proposed algorithm is explained in "[Multi-objective optimization](#)". Computational experiments are presented and analyzed in "[Comparative methods](#)", and finally, the summary of conclusions is explained in "[Computational experiments](#)".

Literature review

The first article where location and routing decisions were simultaneously studied dates back to 1968 and early 1980s. One of the first author group to analyze a LRP was Karp et al. (1972); few surveys on location-routing problems have been presented in the literature.

A good recent review on it can be found in Prodhon and Prins (2014), but several efforts have been published by Lopes et al. (2016), Laporte (1988), Gao et al. (2016), Zhalechian et al. (2016), Min et al. (1998). For detailed information about classification for the LRP; Laporte (1988) is the first researcher who classifies the LRP models. Min et al. (1998) proposed a classification for the LRP based on the solution methods, and the problem perspective, such as the number of depots, the presence of capacity, the form of the objective function, etc. Nagy and Salhi (Vidović et al. 2016), is based on the LRP models, solution approaches and application areas.

In this paper, we consider LRP with time window (LRPTW) in multi-echelon reverse logistic network which is a general case of the LRP by considering time window for vehicles while picking up demands of each customer. Some recent articles of LRPTW are Fazel Zarandi et al. (2013), Govindan et al. (2014). The terms multi-echelon or NE-echelon VRP/LRP are, in fact, first used in (Gonzalez Feliu et al. 2008). There are only a few papers on systems with more than two echelons in LRP/CLRP. Lee et al. (2010) study a three-echelon LRP with routing decisions on the first and third echelon. They consider capacitated facilities on levels 1–3 and fixed costs for opening facilities on levels 1 and 2. Two MIP models are developed. They consider the routing problems on echelons 1 and 3. A heuristic algorithm is presented by them. Contardo et al. (2012) introduced two algorithms to address the two-echelon CLRP (2E-CLRP). They proposed a branch-and-cut algorithm based on the new formulation, and a new adaptive large-neighborhood search (ALNS) meta-heuristic with the objective of finding good-quality solutions



quickly. But there are many papers with two or more than two echelons in RLN. Krikke et al. (1999) designed a MILP model for a two-stage reverse logistics network of a copier manufacturer. In this model, the processing costs of returned products and inventory costs are considered in the objective function.

Travakkoli-Moghaddam et al. (2013) considered a single-sourcing network design problem for a three-level supply chain. Their model considered risk-pooling, the inventory existence at distribution centers (DCs) under demand uncertainty. Yousefikhoshbakht and Khorram (2012) presented a hybrid two-phase algorithm called sweep algorithm (SW) and ant colony system (ACS) for the classical VRP. At the first stage, the VRP is solved by the SW, and at the second stage, the ACS and 3-opt local search are used for improving the solutions.

Reverse logistics network design includes determining numbers, locations, and capacities of collection, recovery, and disposal centers, and the quantity of flow between them. Reverse logistics networks have special characteristics such as important role of collection/inspection centers that we consider in our LRP. Since return products have different qualities, they have different potentials for recovery activities, too. After testing in collection/inspection centers, return products are divided into recoverable and scrapped products to prevent excessive transportation and to ship the return products directly to proper facilities. Aras et al. (2008) develop a nonlinear model for determining the locations of collection centers in a simple reverse logistics network. The important point regarding their article is the capability of the presented model in determining the optimal buying price of used products with the objective of maximizing the total profit. They developed a heuristic approach based on tabu search to solve the model. Patia et al. (2008) proposed a mixed integer goal programming (MIGP) model to assist in proper design of a multi-product paper recycling logistics network. The model studies the interrelationship between multiple objectives of a recycled paper distribution network. The considered objectives are reduction in reverse logistics cost; product quality improvement through increased segregation at the source.

Within the literature reviewed, some works that considered a multiple objective approach for the LRP were found. Caballero et al. (2007) studied a capacitated LRP to locate a given number of incineration plants for solid animal waste in different cities. Five objectives must be minimized: (1) the total cost of the routes, (2) the total opening cost of selected plants, (3) a social rejection measure based on the number of inhabitants in the cities traversed by the routes, (4) an equity criterion (the maximum social rejection over the set of cities), and (5) another social rejection measure taking the distances between

incineration plants and cities into account. They used an adaptative memory procedure (MOAMP) for the resolution of multi-objective combinatorial problems (MOCO). Lately, Hua-Li et al. (2012) presented a bi-level linear programming model for a bi-objective CLRP with a time window to rescue each customer, raised by emergency situations at the city level. Two objectives are considered. The first one is minimization of total cost and the second one is maximizing service level. A genetic algorithm is proposed to solve the problem.

Samanlioglu (2013) proposed a mathematical model for a three-objective (one economic and two social criteria) two-stage LRP, for an industrial hazardous waste management system in a region of Turkey. They considered recycling and disposal centers in their network. They used a lexicographic weighted formulation to obtain 16 different representative Pareto-optimal solutions.

Applications and numerous solution methods varying from exact to heuristic and metaheuristic approaches have been proposed to solve the LRP. Among many solution procedures, only a few of them are presented here, as follows. Berger et al. (2007) consider the uncapacitated LRP with route length constraints in their study, and they propose a branch-and-price algorithm to solve the problem. The heuristic proposed by Barreto et al. (2007) begins by clustering customers according to the capacity of the vehicles. Then, for each cluster, a TSP is solved—optimally for small clusters and heuristically, using the savings method and 3-opt, for large clusters. Finally, depot locations are found by treating each tour as a single customer. They considered integration of several hierarchical and non-hierarchical clustering methods in addition to several proximity measures to solve the General deterministic LRP. They compared the results of running their procedure on standard LRP datasets, and results were analyzed. Different metaheuristic approaches have also been proposed in the literature to solve larger LRPs. Prins et al. (2006) proposed a memetic algorithm with population management (MAIPM) to solve the LRP with capacitated routes and depots. MAIPM is a very recent form of memetic algorithm in which the diversity of a small population of solutions is controlled by accepting a new solution if its distance to the population exceeds a given threshold. Yu et al. (2010) implement a simulated annealing heuristic (SA) for the CLRP. Each solution is encoded as a list containing one sublist per depot. Each sublist begins with the index of the depot, followed by its routes separated by dummy zeros. The random moves performed are: relocations of a node, exchanges of two nodes, and 2-opt moves. The nodes involved can be customers, dummy zeros and, in relocations and exchanges, depot nodes. These nodes may belong to the same route, to two routes rooted at the same depot, or to routes from distinct depots. The 2-opt moves

are restricted to nodes visited from the same depot. An iterative local search (ILS) by embedding it in a genetic algorithm (GA) is described by Derbel et al. (2012) to solve LRPs. The result GA-ILS is a kind of memetic algorithm in which the local search procedure is replaced by ILS. In another attempt to solve LRPs with metaheuristics, Rath and Gutjahr (2011) consider a three-objective warehouse location-routing problem (WLRP) to establish a supply system after a disaster. This WLRP is a kind of two-echelon LRP with plants, warehouses to be located, and customers. The aim is to minimize the strategic cost (total opening cost of warehouses), to minimize an operative cost (transportation costs from plants to depots and warehousing costs proportional to the throughput of each open warehouse) and to maximize a service measure (total demand satisfied). A metaheuristic based on the epsilon-constraint method is used to compute the Pareto frontier. Each single-objective problem is solved by a metaheuristic based on a mixed integer formulation. Constraints are generated on demand by a variable neighborhood search algorithm and stored in a constraint pool. Results are compared to those obtained by a direct resolution of the mixed integer program (on small instances) or by the classical NSGA-II metaheuristic (on larger instances). They indicate that the proposed solution method gives very good solutions.

Problem description and mathematical formulation

The reverse logistics network discussed in this paper is a multistage logistics network including customers, collection/inspection, recycling, and disposal centers with limited capacities. As shown in Fig. 1, returned products are collected from customer zones into collection/inspection centers and after inspection; they are divided into recoverable products and scrapped products. The recoverable products are carried to the recycling centers, and scrapped products are sent to the disposal centers. The authors implement a CLRPTW in a multi-echelon reverse logistics network. The location-routing problem (LRP) can be defined in this paper as follows. An applicative set of

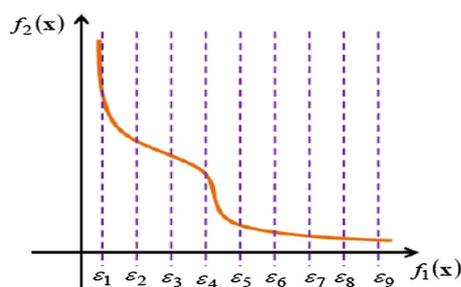


Fig. 1 Pareto diagram

potential collection/inspection, and recycling centers locations and amount of return product of each customer is given. The LRP is to determine the location of facilities and the vehicle routes from facilities to customers to satisfy the objectives of the given problem. The objective of this location-routing problem in RLN is to choose the location and to determine the number of collection/inspection and recycling centers and to determine the quantity of flow between the network facilities.

Problem assumption

- All of the returned products from clients must be collected at the collection centers.
- Each vehicle starts at a depot, visits a set of customers on a route, and returns to the same depot.
- Each customer is served by one vehicle in exactly one turn and should be assigned to only one open collection/inspection centers.
- Locations of customers and disposal centers are fixed and predefined.
- The total returned products of clients on each route is less than or equal to the capacity of the vehicle assigned to that route.
- The sum of the returned products of the customers assigned to each collection/inspection center must not exceed its corresponding capacity.
- Recycling centers have limited capacity.
- A single type of product is considered.
- Soft time windows are considered.
- Fleets of all vehicles are heterogeneous.
- Routing problem is considered only for first echelon.
- First echelon trip must begin/end at the same open collection/inspection centers.
- A predefined percent of demand from each customer is assumed as returned products from the corresponding customer.
- A predefined value is determined as an average scrap fraction.

Model parameters

The following notation is used in the formulation of the capacitated location-routing problem with time window in reverse logistic network design (CLRPTW-RLND) model.

Sets

- I Set of the candidate points for hybrid collection/inspection centers, $\forall i \in I$
- E Set of the candidate points for recovery centers, $\forall e \in E$
- S Fixed set of points for disposal centers $\forall s \in S$
- J Fixed set of points for customer centers, $\forall j, k \in J$



V Set of vehicle routes, $\forall v \in V$
 N Set of turn of customer $\forall n \in N$

Parameters

d_j Demand of customer zone j
 ra_j Rate of return percentage from customer zone j
 θ Average scrap fraction
 cas_v Capacity of vehicle of type v
 caf_i Maximum capacity of hybrid collection/inspection center i
 cat_e Maximum capacity of recycling center e
 g_i Fixed cost of establishing hybrid collection/inspection center i
 f_e Fixed cost of establishing recovery center e
 h_v Fixed cost of using each vehicle that is operated in 1st-echelon for vehicles of type of v
 ϕ_e Recycling cost per unit of product at recycling center e
 ct_{ij} Transportation cost per unit of returned products from customer zone j to collection/inspection center i
 cr_{ie} Transportation cost per unit of recoverable products from hybrid collection/inspection center i to recovery center s
 cd_{is} Transportation cost per unit of scrapped products from hybrid collection/inspection center i to disposal center s
 t_{ijv} Travel time between node i and j with vehicle v
 a_j Earliest arrival time to customer j of the soft time window
 b_j Latest arrival time to customer j of the soft time window
 ue_j Lower bound of the soft time window for customer j
 ul_j Upper bound of the soft time window for customer j
 pe Earliest penalty cost for hybrid collection/inspection center (penalty cost of one unit earliest)
 pl Lateness penalty cost for hybrid collection/inspection center (penalty cost of one unit lateness)

Note: In this paper, we consider one of the time variants where the time constraint is ‘soft’, that is, it can be violated because of the model constraints; in other words, the classical vehicle routing problem with time windows (VRPTW) is an extension of the VRP where the service at each customer must take place within a given time interval (hard time window). The latter is often relaxed in practice (leading to soft time window) which enables early and late servicing with some penalty costs. A soft time window with non-negative boundaries (a_j, b_j) is then defined for each node j . The time window at the depot (ue_j, ul_j) can be thought of as the scheduling horizon of the problem.

Decision variable

q_{ijv} Quantity of returned products shipped from beginning of route from hybrid collection/inspection center i to node j in vehicle route v (load remaining in vehicle v after leaving node j when Starting from collection/inspection center i)
 q_{iv} Quantity of load of the vehicle v after leaving hybrid collection/inspection center i
 p_{iev} Quantity of recoverable products shipped from hybrid collection/inspection center i to recovery centers by vehicle v
 o_{isv} Quantity of scrapped products shipped from hybrid collection/inspection center i to disposal centers by vehicle v
 qT_i Total amount of return product collected to the hybrid collection/inspection center i
 pT_e Total amount of recoverable products shipped to the recovery center e
 ot_s Total amount of scrapped products shipped to the disposal center s
 TC_{max} Maximum time for completion of the collecting return products
 TF_v Time for completion of the collection by vehicle v
 w_{jv} Starting time of the service to customer j by vehicle v
 w_{jv}^e The amount of earliest time of the starting service to customer j by vehicle v
 w_{jv}^l The amount of Latest time of the starting service to customer j by vehicle v
 z_{ij} Binary variable which is 1 if customer j is allocated to hybrid collection/inspection centers i and zero otherwise
 Y_i Binary variable which is 1 if hybrid collection/inspection center is opened at location i and zero otherwise
 Y_e Binary variable which is 1 if recovery center is opened at location e and zero otherwise
 x_{ijnv} Binary variable which is 1 if vehicle v goes from hybrid collection/inspection center i to customer j on turn of n and zero otherwise
 R_{jkv} Binary variable which is 1 if customer k immediately precedes customer j by vehicle v and zero otherwise
 $c1_v, c2_v$ Binary variable which is 1 if vehicle v is used in first and second level and zero otherwise
 as_{iv} Binary variable which is 1 if vehicle v is allocated to hybrid collection/inspection center i and zero otherwise

as'_{iev} Binary variable which is 1 if goes from hybrid collection/inspection center i to recovery center e by vehicle v and zero otherwise

as''_{isv} Binary variable which is 1 if goes from hybrid collection/inspection center i to disposal center s by vehicle v and zero otherwise

In terms of the above notations, the capacitated location-routing problem with time window in reverse logistic network design (CLRPTW-RLND) can be formulated as follows:

Mathematical model

$$\begin{aligned} \min f_1(x) = & \sum_{i \in I} g_i Y_i + \sum_{e \in E} f_e Y_e + \sum_{v \in V} h_v (c1_v + c2_v) \\ & + \sum_{i \in I} \sum_{j \in J} \sum_{v \in V} ct_{ij} q_{ijv} + \sum_{i \in I} \sum_{e \in E} \sum_{v \in V} (cr_{ie} + \varphi_e) p_{iev} \\ & + \sum_{i \in I} \sum_{s \in S} \sum_{v \in V} cd_{is} o_{isv} + \sum_{j \in J} \sum_{v \in V} (pe \times we_{jv} + pl \times wl_{jv}) \end{aligned} \tag{1}$$

$$\min f_2(x) = TC_{\max} \text{ s.t.} \tag{2}$$

$$\sum_{i \in I} z_{ij} = 1 \quad \forall j \in J \tag{3}$$

$$\sum_{j \in J} z_{ij} \leq M \cdot Y_i \quad \forall i \in I \tag{4}$$

$$\sum_{n \in N} \sum_{v \in V} x_{ijnv} = z_{ij} \quad \forall i \in I, j \in J \tag{5}$$

$$\sum_{j \in J} x_{ijnv} \leq 1 \quad \forall i \in I, n \in N, v \in V \tag{6}$$

$$\sum_{j \in J} x_{ijnv} \leq \sum_{j \in J} x_{ijn-1v} \quad \forall i \in I, n \in N, v \in V \tag{7}$$

$$R_{jkv} = \sum_{i \in I} \sum_{n \in N} x_{ijnv} x_{ikn+1v} \quad \forall j, k \in J, v \in V \tag{8}$$

$$q_{ikv} \geq q_{ijv} + rakd_k - M(1 - R_{jkv}) \quad \forall i \in I, j, k \in J, v \in V \tag{9}$$

$$q_{ikv} \leq cas_v \quad \forall i \in I, k \in J, v \in V \tag{10}$$

$$q_{i1v} = 0 \quad \forall i \in I, v \in V \tag{11}$$

$$qt_i = \sum_{j \in J} \sum_{n \in N} \sum_{v \in V} ra_j d_j \times x_{ijnv} \quad \forall i \in I \tag{12}$$

$$qt_i = \sum_{e \in E} \sum_{v \in V} p_{iev} + \sum_{s \in S} \sum_{v \in V} o_{isv} \quad \forall i \in I \tag{13}$$

$$(\theta) qT_i = \sum_{e \in E} \sum_{v \in V} p_{iev} \quad \forall i \in I \tag{14}$$

$$pT_e = \sum_{i \in I} \sum_{v \in V} p_{iev} \quad \forall e \in E \tag{15}$$

$$OT_s = \sum_{i \in I} \sum_{v \in V} o_{isv} \quad \forall s \in S \tag{16}$$

$$qt_i \leq caf_i Y_i \quad \forall i \in I \tag{17}$$

$$PT_e \leq cat_e Y_e \quad \forall e \in E \tag{18}$$

$$\sum_{i \in I} as_{iv} \leq c_v \quad \forall v \in V \tag{19}$$

$$\sum_{j \in J} \sum_{n \in N} x_{ijnv} \leq BM \times as_{iv} \quad \forall i \in I, v \in V \tag{20}$$

$$\sum_{e \in E} as'_{isv} + \sum_{s \in S} as''_{isv} \leq as_{iv} \quad \forall i \in I, v \in V \tag{21}$$

$$cas_v \times as''_{isv} \geq o_{isv} \quad \forall i \in I, s \in S, v \in V \tag{22}$$

$$cas_v \times ca'_{iev} \geq p_{iev} \quad \forall i \in I, e \in E, v \in V \tag{23}$$

$$TC_{\max} \geq TF_v \quad \forall v \in V \tag{24}$$

$$TF_v = \sum_{i \in I} \sum_{j \in J} \sum_{n \in N} t_{ijv} \times x_{ijn=1v} + \sum_{j,k \in J} R_{jkv} \times t1_{jkv} \quad \forall v \in V \tag{25}$$

$$w_{jv} \geq ue_j \sum_{i \in I} \sum_{n \in N} x_{ijnv} \quad \forall j \in J, v \in V \tag{26}$$

$$w_{jv} \leq ul_j \sum_{i \in I} \sum_{n \in N} x_{ijnv} \quad \forall j \in J, v \in V \tag{27}$$

$$w_{jv}^e \geq a_j - w_{jv} - M \left(1 - \sum_{i \in I} \sum_{n \in N} x_{ijnv} \right) \quad \forall j \in J, v \in V \tag{28}$$

$$w_{jv}^l \geq w_{jv} - b_j - M \left(1 - \sum_{i \in I} \sum_{n \in N} x_{ijnv} \right) \quad \forall j \in J, v \in V \tag{29}$$

$$w_{kv} \geq w_{jv} + t1_{jkv} - M(1 - R_{jkv}) \quad \forall j, k \in J, v \in V \tag{30}$$

$$L_{i,j,k,n,v} = x_{ijnv} \times x_{ikn+1v} \quad \forall i \in I, j, k \in J, n \in N, v \in V \tag{31}$$

$$L_{i,j,k,n,v} \leq x_{ijnv} \quad \forall i \in I, j, k \in J, n \in N, v \in V \tag{32}$$

$$L_{i,j,k,n,v} \leq x_{tikn+1v} \quad \forall ti \in I, tj, k \in tJ, n \in N, v \in V \tag{33}$$

$$L_{i,j,k,n,v} \geq x_{ijnv} + x_{ikn+1v} - 1 \quad \forall i \in I, j, k \in J, n \in N, v \in V \tag{34}$$

$$z_{ij}, Y_e, Y_i, x_{ijnv}, R_{jkv}, c_v, as_{iv} \in \{0, 1\} \quad \forall i \in I, j, k \in J, e \in E, n \in N, v \in V \tag{35}$$

$$\begin{aligned}
 & q_{ijv}, P_{iev}, o_{isv}, qt_i, pt_e, ot_s, tc_{max}, TF_v, W_{jv}, w_{jv}^1, w_{jv}^e, \\
 & L_{ijknv} \geq 0 \quad \forall i \in I, j, k \in J, e \in E, n \in N, v \in V
 \end{aligned}
 \tag{36}$$

Objective function (1) minimizes the total cost consisting of the sum of the fixed hybrid collection/inspection center location costs, the fixed recovery center location costs, the fixed costs of employing vehicles, variable transportation and processing costs and penalty cost, when arrival time for a node is not in the determined time window, in first echelon. Objective function (2) minimizes the maximum time of completion of the collecting return products. Constraint (3) ensures that each customer must be assigned to a collection/inspection center. Constraint (4) ensures that the customer can be assigned to hybrid collection/inspection center if and only if it is open. Constraints (5)–(8) are related to turn of the customer for collecting their return products. Constraint (9) calculates the vehicle load of each vehicle after finishing the service to each customer. Constraints (10) require that the capacity of each vehicle should be respected. Constraint (11) shows the initial load of each vehicle that is equal to zero. Constraint (12) ensures that the returned products of all customers are collected. Constraints (13)–(16) assure the flow balance at hybrid collection/inspection, recovery and disposal centers. Equations (17) and (18) are capacity constraints on hybrid collection/inspection and recovery centers, respectively. Constraint (19) ensures that each vehicle is allocated to one hybrid collection/inspection. Constraint (20) and (21) ensure that vehicle v is used in first and second level if it is allocated to hybrid collection/inspection center. Constraint (22) and (23) are related to capacity of vehicles in first and second level. Constraint (24) is related to total collected time. Constraint (25) calculates the collected time for each vehicle along a route. Equations (26)–(30) are related to time window Constraints. Constraints (31)–(34) are a set of constraints introduced to convert Constraints (8) to a linear form.

The Constraints (8) is in a nonlinear form; therefore, another binary variable along with a set of constraints is introduced to convert it to a linear form. The transformation equation (Eqs. 31–34) is added to the original constraints. The above equation means that the new variable ($L_{i,j,k,n,v}$) is also a binary one. Finally, Constraints (35) and (36) enforce the binary and non-negativity restrictions on decision variables.

Multi-objective optimization

Many real-world problems involve simultaneous optimization of several objective functions. Generally, optimization in terms of the number of objective functions and optimization criteria is divided into two categories of single-objective

optimization problems and multi-objective optimization problems. A single-objective optimization algorithm is terminated upon obtaining an optimal or near-optimal solution. The purpose of solving a single-objective problem is to improve the unique performance index that should be the minimum or maximum. But in some cases, the problem may have more than one objective that is called multi-objective optimization problems. And because of the conflict between the objectives, usually by improving the value of one of the objectives, the other one becomes worse. So, it is natural to find a set of solutions depending on the non-dominance criterion, because it is difficult to find a single solution for a multi-objective problem. Solutions which dominate the others but do not dominate themselves, are called non-dominated solutions. When we have a globally optimal solution that is not dominated by any other solution in the feasible space, it is called Pareto optimal. The set of all Pareto-optimal solutions is also termed the Pareto-optimal set or efficient set. Their corresponding images in the objective space are called the Pareto-optimal frontier. There exists various algorithms such as heuristic and metaheuristic, for optimizing the multi-objective optimization problems.

Comparative methods

In this section, we have designed two solution methods, one of them is metaheuristic procedure based on the non-dominated sorting genetic algorithm II (NSGA-II) for small and large test problem, and the other is exact procedure based on the ϵ -Constraint Method. We try to evaluate the efficiency of the suggested a famous multi-objective evolutionary algorithms (EAs), namely NSGA-II with ϵ -Constraint Method. NSGA-II is coded using MATLAB software and run on a personal computer with 2.4 GHZ CPU Intel Core i7 Duo processor and 2.00-GB of RAM memory and ϵ -Constraint method is coded using GAMS software.

Procedure based on ϵ -constraint method

We solve the presented bi-objective model by the ϵ -constraint method in the GAMS software using Baron solver for the given small test problem. There is a conflict between two objectives. It means that the units of our two objectives are minimizing cost and time, respectively. Logically, if we incur more cost, we have less time and vice versa.

This method is based on optimizing one of the most preferred objective functions, and considering the other objectives as constraints. The authors provide some basic definitions to better understand the ϵ -Constraint Method. Without loss of generality, let us assume the following multi-objective minimization problem (MOMP) with m Objectives (Mavrotas 2009):

$$\begin{aligned}
 &\min (f_1(x), f_2(x), \dots, f_m(x)) \\
 &\text{s.t.} \\
 &g(x) \leq 0, \quad h(x) = 0 \\
 &x \in s,
 \end{aligned} \tag{37}$$

where x is the vector of decision variables and S is the feasible region. As we said, in this method, we optimize one of the objective functions using the other objective functions as constraints, as shown below:

$$\begin{aligned}
 &\min f_1(x) \\
 &\text{s.t.} \\
 &f_2(x) \leq e_2, \\
 &f_3(x) \leq e_3, \dots \\
 &f_n(x) \leq e_n, \\
 &g(x) \leq 0, \quad h(E) = 0, \\
 &x \in s.
 \end{aligned} \tag{38}$$

By parametrical variation in the rRight hand side (RHS) of the constrained objective functions (e_n), we can obtain the efficient solutions of the problem and draw Pareto diagram (Fig. 1). One of the advantages of the ε -constraint in each run is to produce a different efficient solution, thus obtaining a more rich representation of the efficient set.

Procedure based on non-dominated sorting genetic algorithm II (NSGA-II)

NSGA-II is one of the most well-known multi-objective optimization evolutionary algorithms. It is basically a genetic algorithm with special characteristics in the

selection phase. NSGA, for the first time, was introduced by Deb and Srinivas (1995), but because of some of the disadvantages, such as computational complexity, time-consuming and inadequacy of this edition, the second edition abbreviated NSGA-II was developed by Deb et al. (2002). The main features of this algorithm are: fast non-dominated sorting approach, fast crowded distance estimation procedure, simple crowded comparison operator and binary tournament selection operator (Tavakkoli-Moghaddam et al. 2012). Generally, the principal components of the NSGA-II procedure are summarized below.

Population initialization

An initial parent population p_0 of size number of population (n_{pop}) is generated randomly based on the problem range and constraint. A series of genes that arrange sequentially is called chromosome. The number of genes in a chromosome is equal to the number of decision variables. Chromosome description is one of the most significant parts of the algorithm that is taken into account as the code form. In this paper, there-position is used for chromosome. The first position shows the allocation of customers to the hybrid collection/inspection centers and allocation of first level’s vehicles that is formed with a matrix dimensions $I*(J + V)$ (the number of rows = number of hybrid collection/inspection center and the number of columns = number of customers + number of vehicle). The second part shows the allocation of recycling center to the hybrid collection/inspection centers and allocation of second level’s vehicles that is formed with a matrix

		j					v				
		1	2	3	4	5	1	2	3	4	
Position 1	I	1	0.94	0.07	0.64	0.37	0.38	0.12	0.29	0.34	0.47
		2	0.78	0.52	0.14	0.11	0.45	0.25	0.21	0.62	0.53
		3	0.49	0.91	0.14	0.37	0.15	0.68	0.45	0.30	0.51

		e		v				
		1	2	1	2	3	4	
Position 2	I	1	0.77	0.22	0.39	0.70	0.21	0.86
		2	0.46	0.38	0.53	0.21	0.63	0.61
		3	0.50	0.68	0.50	0.43	0.47	0.41

		s	v				
		1	1	2	3	4	
Position 3	I	1	0.60	0.74	0.43	0.27	0.52
		2	0.83	0.60	0.70	0.59	0.88
		3	0.58	0.27	0.33	0.20	0.29

Fig. 2 Structure of a chromosome for algorithm NSGA-II

dimensions $I^*(E + V)$. Finally, the third part shows the allocation of disposal centers to the hybrid collection/inspection centers and allocation of second level's vehicles that is formed with a matrix dimensions $I^*(S + V)$. Moreover, Element of the matrix or each gene value (allele) of the chromosome is generated randomly of real values in the range (0, 1). Figure 2 depicts the structure of a chromosome.

Chromosome structure consists of three sections. Since most of the constraints and variables are related to each other sequentially, the following procedures are applied for constraint handling.

There are three types of variables: location, routing and allocation, and some continuous variables. Location variables are first stage. In other words, location decisions are selected randomly earlier according to capacity constraints. In this way, solutions will be generated feasibly. Also, allocation and routing decisions are made based on the previous location decisions. Therefore, nodes are assigned to previous stage nodes according to capacities and then, a random routing solution is selected. By this way, this part of solution is feasible. Finally, continuous variables are computed sequentially. In this way, if a variable such as time window be infeasible, then the objective function will be penalized.

Non-dominated sorting

Before selection is performed, every individual (chromosome) in the population is ranked based on the non-domination sorting procedure to create Pareto fronts. All non-dominated individuals are classified into one category in such a way that each individual of the population under evaluation obtains a rank equal to its non-domination level (1 is the best level, 2 is the next-best level, and so on), where the Front one consists of all solutions with the smallest rank, that are not dominated by any other solutions. The second front is made by all solutions that only dominated by solutions in front number one. A multi-objective model has n objective functions, solution x and y are placed in same front when do not dominate each other, a solution x dominate y when the following conditions are successful:

- For all the objective functions, solution x is not worse than another solution y .
- For at least one of the n objective functions, x is strictly better than y .

Crowding distance

Crowding distance proposes an estimate of the density of solutions surrounding a particular solution. The individuals in population at the first time are selected based on rank

and the member with the lower rank is chosen, but if two solutions have the same rank, the remainder of the population is selected based on crowding distance between members. So, the member with the larger crowding distance is selected if they share an equal rank. The crowding distance used in the NSGA-II is computed by the following equation,

$$d_i^j = \sum_{i=1}^n \frac{f_j^{i+1} - f_j^{i-1}}{f_j^{\max} - f_j^{\min}}, \tag{39}$$

where f_j^{\min} and f_j^{\max} are, respectively, the minimum and maximum values of the objective function j in the population, f_j^{i+1} is the value of the objective function j of the $(i + 1)$ -th solution and f_j^{i-1} is the value of the objective function j of the $(i - 1)$ -th solution.

Crossover

Crossover operator combines characteristics of parent chromosomes and generates new solutions called offspring (children) by changing some part of parent chromosomes. The idea behind crossover is that the new chromosome may be better than both of the parents if it takes the best characteristics from each of the parents with crossover rate p_c that is usually consider $0.5 < p_c < 1$. There are several ways for crossover operator, including: one point crossover, partially mapped crossover (PMX), ordered crossover (OX), Cycle crossover (CX), and Arithmetic crossover. Generally, it chooses two parent chromosomes from a population based on the crowding selection operator described, with a crossover chance, crossover this parents to form new offspring through mating pairs of chromosomes. Next, a new population of offspring with a size of n is created. We can use the selection, the crossover, and the mutation operators to create a population consisting of the current and the new population of the size of $(n_{pop} + n)$. Proper crossover operator called arithmetic crossover is utilized in this paper, arithmetic crossover operator linearly combines two parent chromosome vectors to produce two new offspring according to the equations: (α is a random weighing factor chosen before each crossover operation). Figure 3 depicts the graphical representation of arithmetic crossover with $\alpha = 0.5$.

$$\begin{aligned} y_{2i} &= \alpha_i x_{2i} + (1 - \alpha_i) x_{1i} \\ y_{1i} &= \alpha_i x_{1i} + (1 - \alpha_i) x_{2i} \end{aligned} \tag{40}$$

Mutation

Unary variation operator in genetic algorithm is named mutations. After generating the children from the crossover

Fig. 3 shows a graphical representation of Arithmetic crossover with $\alpha = 0.5$

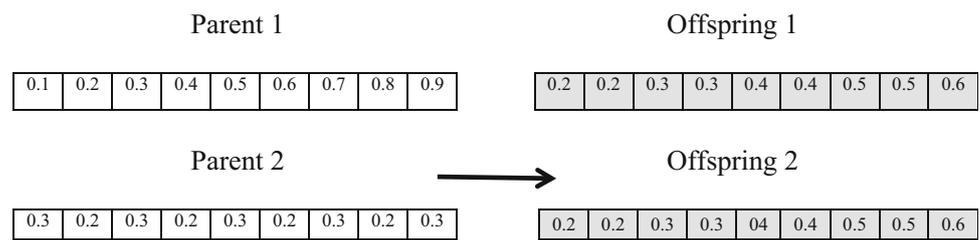


Table 1 Uniform distribution for different parameters in each test problem

	j									
Value and distribution	5	22	$U \sim (0.2, 0.4)$	$U \sim (50, 200)$	$U \sim (0.7, 0.8)$	$U \sim (100, 500)$	$U \sim (100, 500)$	$U \sim (1, 5)$		
	8	25								
	10	30	$U \sim (3000, 20000)$	$U \sim (3000, 20000)$	$U \sim (10, 30)$	$U \sim (100, 400)$	$U \sim (30, 90)$	$U \sim (30, 90)$		
	12	40								
	18	50								
	20		$U \sim (1, 5)$	$U \sim (10, 12)$	$a(j) \times 0.8$	$b(j) \times 1.2$	$U \sim (10, 30)$	$U \sim (10, 30)$		

operator, mutations takes place. Mutations operator as well as the crossover operator creates a new space for searching in the algorithm. To enhance the diversity of a newly generated population, a mutation operator comes to the picture at the time of movement from the current population to the new population to explore new solution spaces. Using a mutation operator, a few genes of a candidate chromosome are randomly selected to change their values based on a predetermined mutation probability of pm. Mutation helps to prevent the population from stagnating at any local optima. Based on a permutation encoding, there are different mutation operators such as random resetting mutation, scramble mutation, flip bit, and uniform, inversion mutation, and mutation for decimal number, shift, and swap that can generate neighborhoods of a current solution. In this paper, we employ the mutation operator for decimal number in solution algorithm. In this case, in which the value of each gene replaced with one of the values in the range of L_i (lower bound) to U_i (upper bound) randomly. The mutation, in Eq. (35) is shown.

$$\langle x_1, \dots, x_n \rangle \rightarrow \langle x'_1, \dots, x'_n \rangle ; x_i, x'_i \in [L_i, U_i] \tag{41}$$

Recombination and selection

In steps of the NSGA-II algorithm, there are two steps that the algorithm should do selection. First selection of individuals is carried out using a binary tournament selection

Table 2 Factors and their levels in the NSGA-II algorithm for small (S) and large (L) sizes

Factors	Level	
	S	L
pop_size	50	150
P_c	0.4	0.35
p_m	0.7	0.8
Maxit	100	200

Table 3 Default value for different parameters in each test problem

Parameters	Default values
Number of hybrid collection/ inspection center	2, 3, 4, 5, 7, 8, 10, 12, 15
Number of recycling center	1, 2, 3, 4, 5, 7, 9, 10
Number of disposal centers	1, 2, 3, 4, 5, 6
Number of customers	5, 7, 8, 10, 12, 15, 17, 18, 20, 22, 25, 30, 40, 50
Number of vehicle	2, 3, 4, 5, 8, 10, 12, 15, 18, 20
Number of turn of customer	5, 7, 8, 10, 12, 15, 17, 18, 20, 22, 25, 30, 40, 50

operator. The binary tournament selection strategy repeatedly selects one parent regarding n from each pair of two randomly selected individuals until all n_{pop} parents have been selected. These parent individuals are then paired randomly. As we said with crossover probability p_c , each pair is recombined by the crossover operator to create two

child individuals that is enter the offspring population; Otherwise the two parents undergoes a mutation by the mutation operator with mutation probability pm and then enter the offspring population so we have new generated solutions. Afterward, the current population and new generated solutions are combined together. Then sorting is performed using the value of the non-dominance and using the crowding distance. If two populations are from different fronts, the lowest front number is selected and if they are belong to the same front, the solution with the highest crowding distance is selected to form a mating pool. Finally, a population of an exact size of n_{pop} is obtained using the sorting procedure. The new population is used to generate the next new offspring by repeating the above steps in order. This process is repeated until the stopping condition is met. The stopping criterion considered is a fixed number of iterations.

Computational experiments

This section presents a fair comparison between the two solution methods, to compare the numerical results generated by the proposed NSGA-II and ϵ -Constraint proposed here. For this target, 2 problem groups, plans one in small sizes, and other in medium and great sizes. First smaller sample problem solved by NSGA-II and resulted solutions compare by ϵ -constraint method with resulted solutions from model solving.

Parameter values

Parameters in these test problems have different values, but in a defined tolerance. Table 1 shows uniform distribution for parameters.

Parameter tuning

To evaluate the performance of the algorithm, we set the assigned value of the algorithm parameters. Parameter values have a very significant effect on the efficiency of the algorithm. If these values are not set truly, getting the proper result will become difficult. To tune the parameters of algorithms, we consider two different sets of problems, namely small sizes (i.e., problem numbers 1–10) and large sizes (i.e., problem number 10–20). To determine the values of parameters, we use the Taguchi methodology. In this section, each algorithm is run for different combinations of parameters and their levels. In the Taguchi methodology, we need an index to compare different combinations of parameters. The aim of this method for designing parameters was reaching a mean value of the objective function and reduces variation in the response variable. Table 2

Table 4 Comparison between the performances of ϵ -Constraint and the proposed NSGA-II (optimality and runtime) for small instances

Problem number	Indices										Comparison (ϵ -constraint, NSGA-II)				
	ϵ -Constraint					NSGA-II					Gap Objective		Gap Objective		
	Customers	Hybrid collection/ inspection centers	Recovery centers	Disposal centers	Vehicles	Turn of customer	Objective function 1	Objective function 2	Runtime	Objective function 1	Objective function 2	Runtime	Objective 1	Objective 2	Time
1	5	2	1	1	2	5	28918.867	37.712	20.451	28918.867	32.712	6.824	1	1	0.333676
2	5	2	2	1	2	5	18791.195	23.123	18.815	18791.195	24.154	5.395	1	1	0.955499
3	5	2	2	1	3	5	32982.012	25.033	25.223	36942.015	29.215	7.236	0.879934	0.832941	0.286881
4	5	3	2	1	3	5	34538.964	23.979	35.258	35221.032	24.879	8.028	0.980252	0.962467	0.227693
5	7	3	2	1	4	7	30055.121	21.209	52.252	31076.115	23.721	10.313	0.966029	0.88156	0.19737
6	7	3	3	2	4	7	40277.534	41.754	68.277	42001.864	45.983	15.929	0.957189	0.898716	0.2333
7	8	4	3	2	3	8	50712.868	23.909	180.215	51614.743	25.541	20.905	0.982216	0.931741	0.116
8	8	5	3	2	3	8	54953.299	21.677	730.63	56983.840	23.267	22.463	0.963049	0.92665	0.030754
9	10	4	3	2	3	10	4601.577	30.187	1323.6	48812.282	31.731	35.034	0.938899	0.948852	0.026469
10	10	4	3	2	4	10	40752.457	25.815	3540.8	42356.520	26.029	34.124	0.960639	0.99171	0.009637

Fig. 4 Comparison of two methods to evaluate solutions based on time and sample

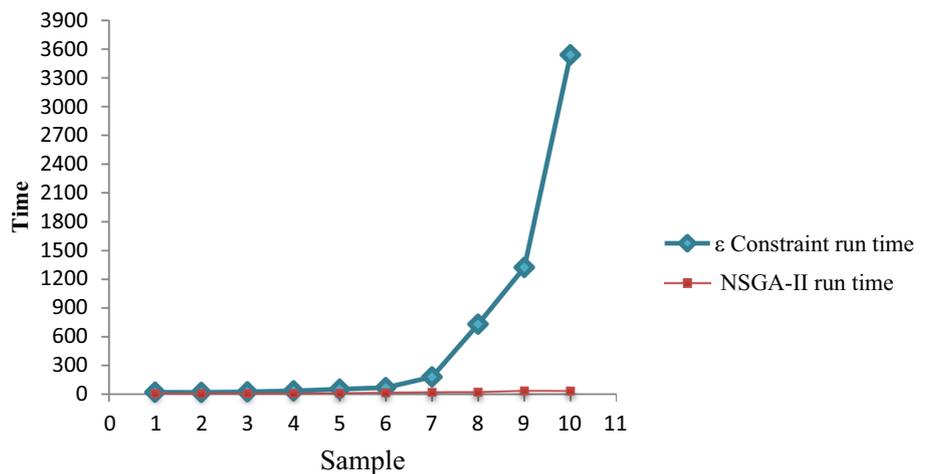


Table 5 Payoff table of problem 5

The objective function	Z ₁	Z ₂
Z ₁	30055.120	21.2089
Z ₂	49538.743	18.4977

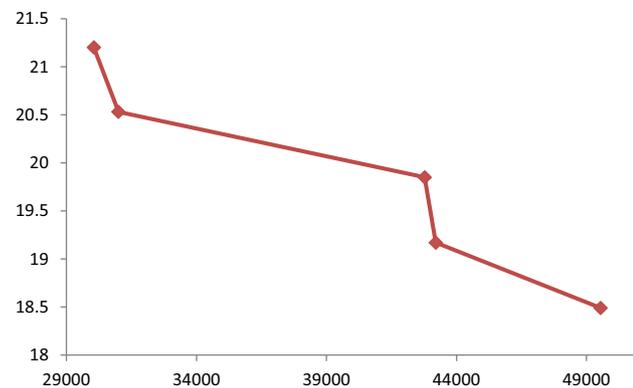


Fig. 5 The Pareto front chart of sample 5 in GAMS Software

shows the parameters and their levels for small and large sizes in the NSGA-II algorithm.

Parameters are selected by trial and error and then best ones are selected.

Table 3 shows the default value for different parameters in each test problem. According to Table 3, different test problems are carried out in Tables 4 and 6.

Comparison results of metaheuristic and exact solution methods

Ten samples were solved by NSGA-II and ε-Constraint in small sizes and were compared in the Table 4. In Table 4,

first column represents problem number, second and third columns are customer index and hybrid collection/inspection centers index, respectively. And columns fourth and fifth show recovery centers and disposal centers indexes, respectively. And column sixth and seventh ones show vehicles index and turn of customer index, respectively. Columns eighth–thirteenth show the value of objective function 1 and objective function 2 with runtime which is resulted from running by ε-Constraint and NSGA-II. Finally, last columns show the amount of error or Gap (Eq. 36) between two methods.

Figure 4 shows that the exact time solution increases exponentially. Comparing the exact time and the time of solving meta-heuristic algorithm from Fig. 4 and Table 4, it can be seen that metaheuristic algorithm has reached optimal solution. This result makes the applicability of the algorithm clear.

The payoff Table 5 and Pareto Fig. 5 which obtains from running Gams software are given for Example 5. Pareto chart shows that with increasing time, total cost is decreasing and vice versa, so they are in conflict with each other. Also, amount of ε for Pareto chart is calculated by Eq. (42).

$$\epsilon = 30055.120 + \frac{(49538.743 - 30055.120)}{5} \times counter \tag{42}$$

Figure 6 shows the optimization processes and solving time. From this figure, it can be said that time chart slope is decreasing. This leads us to the conclusion that metaheuristic algorithm is very applicable.

10 instances or samples of large problems have been solved by NSGA-II algorithm and such as Table 4; the results are shown in Table 6.

Figure 7 shows the Pareto front chart of sample 15 which is obtained from running Matlab software. Pareto

Fig. 6 The descending trend of the relative optimality and runtime ratios

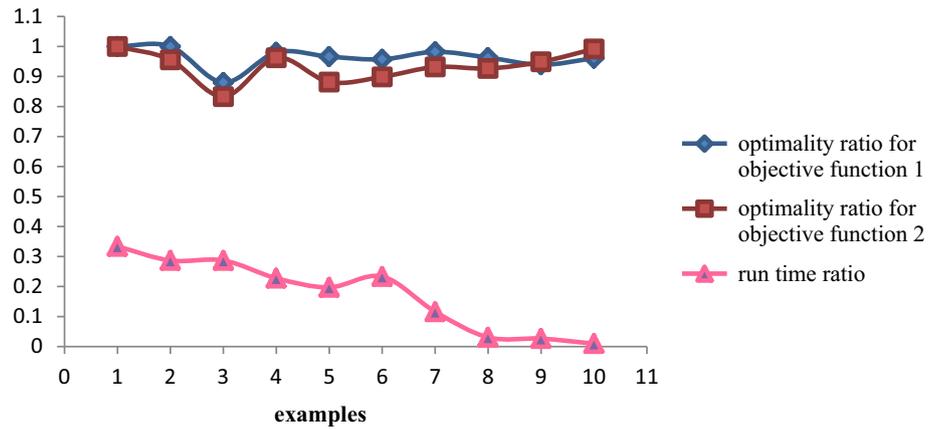


Table 6 Objective function and runtime of the NSGA-II algorithm for large instances

Problem number	Indices						NSGA-II		
	Customers	Hybrid collection/ inspection centers	Recovery centers	Disposal centers	Vehicles	Turn of customer	Objective function 1	Objective function 2	Runtime
11	12	5	4	3	5	12	120511	19	479.4693
12	15	7	4	3	8	15	134135	19	548.8661
13	17	8	5	4	10	17	185473	17	627.1582
14	18	10	7	5	12	18	195446	27	689.5648
15	20	10	9	5	15	20	250450	24	899.1653
16	22	12	9	5	15	22	226345	23	983.3567
17	25	12	9	5	18	25	331672	35	1105.934
18	30	15	10	6	18	30	574537	62	1318.657
19	40	15	10	6	20	40	955241	86	1637.176
20	50	15	10	6	20	50	1356581	106	2584.628

Fig. 7 The Pareto chart of sample 15 in MATLAB software

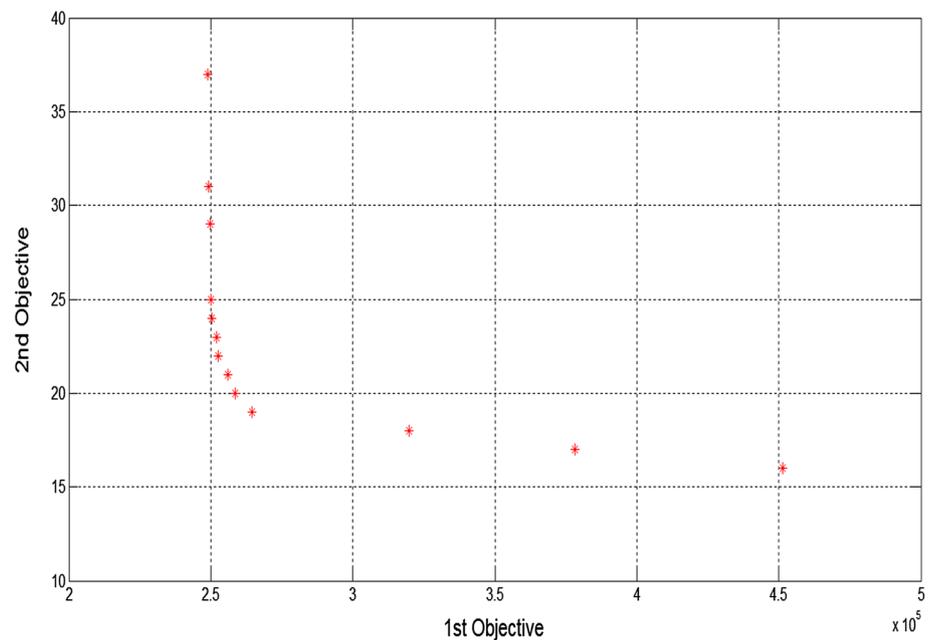


chart shows that with increasing time, total cost is decreasing and vice versa, so they are in conflict with each other.

Conclusions

This paper has presented a new bi-objective location-routing problem (LRP) with attributes such as multi-level in a reverse logistics network, multiple depots, capacitated and heterogeneous fleets of vehicles, soft time windows, and penalty cost. Warning and threat on environment have forced the researchers to notice to the transportation and reverse logistics activities as collecting of expired products can have great effect on the environment seriously and think to application solutions.

The proposed RL network model included hybrid collection/inspection centers, recovery centers and disposal centers. We presented a new bi-objective mathematical programming (BOMP) for LRPTW in reverse logistic. Since this type of the problem was NP-hard, the Non-dominated Sorting Genetic Algorithm II (NSGA-II) was proposed to reach the Pareto frontier for the proposed problem in large sized problems.

At first, we solved problem with ε -Constraint method by GAMS software, but due to the complexity of the problem and available processing facilities, when the size of the instances increased for problems and for dimensions bigger than ten customers, metaheuristic method must, therefore, be applied. A number of test problems have been generated to evaluate the performance of the proposed algorithm (non-dominated sorting genetic algorithm II (NSGA-II)) in comparison with ε -Constraint. The result indicated that NSGA-II worked more efficiently than the ε -Constraint.

For further researches, we can suggest the following notes:

- Authors may focus on handling uncertainties in traveling times and on exploring LRP in a closed-loop supply chain system.
- The multi-periods LRP can be considered.
- Robust optimization technique may be applied to develop model.
- Inventory optimization can be considered through mathematical modeling.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

- Aras N, Aksen D, Tanugur AG (2008) Locating collection centers for incentive-dependent returns under a pickup policy with capacitated vehicles. *Eur J Oper Res* 191:1223–1240
- Barreto S, Ferreira C, Paixao J, Sousa Santos B (2007) Using clustering analysis in a capacitated location-routing problem. *Eur J Oper Res* 179(3):968–977
- Berger R, Coullard C, Daskin M (2007) Location-routing problems with distance constraints. *Transp Sci* 41(1):29–43
- Caballero R, González M, Guerrero FM, Molina J, Paralara C (2007) Solving a multiobjective location routing problem with a metaheuristic based on tabu search. Application to a real case in Andalusia. *Eur J Oper Res* 177(3):1751–1763
- Contardo C, Hemmelmayr V, Crainic TG (2012) Lower and upper bounds for the two-echelon capacitated location-routing problem. *Comput Oper Res* 39:3185–3199
- Deb K, Srinivas N (1995) Multi objective optimization using non-dominated sorting in genetic algorithms. *Evol Comput* 2:221–248
- Deb K, Pratap A, Agarwal S et al (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans Evol Comput* 6(2):182–197
- Derbel H, Jarboui B, Hanafi S, Chabchoub H (2012) Genetic algorithm with iterated local search for solving a location-routing problem. *Expert Syst Appl* 39(3):2865–2871
- Fazel Zarandi MH, Hemmati A, Davari S, Turksen IB (2013) Capacitated location-routing problem with time windows under uncertainty. *Knowl Based Syst* 37:480–489
- Gao S, Wang Y, Cheng J, Inazumi Y, Tang Z (2016) Ant colony optimization with clustering for solving the dynamic location routing problem. *Appl Math Comput* 285:149–173
- Gonzalez Feliu J, Perboli G, Tadei R, Vigo D (2008) The two-echelon capacitated vehicle routing problem. Technical report, Control and Computer Engineering Department, Politecnico di Torino, Italy
- Govindan K, Jafarian A, Khodaverdi R, Devika K (2014) Two-echelon multiple-vehicle location routing problem with time windows for optimization of sustainable supply chain network of perishable food. *Int J Prod Econ* 152:9–28
- Hua-Li S, Xun-Qing W, Yao-Feng X (2012) A Bi-level programming model for a multi-facility location-routing problem in Urban emergency system. In: Zhang L, Zhang C (Eds.) *Engineering Education and Management*, Springer, Berlin, p 75–80
- Hyunsoo K, Jaehwan Y, Kang-Dae L (2009) Vehicle routing in reverse logistics for recycling end-of-life consumer electronic goods in South Korea. *Transp Res Part D* 14(2009):291–299
- Karp R, Miller R, Thatcher J (eds.) (1972). *Reducibility among combinatorial problems*. In: *Complexity of computer computations*, Plenum Press, New York, p 85–104
- Krikke HR, Van Harten A, Schuur PC (1999) Reverse logistic network redesign for copiers. *OR Spektrum* 21:381–409
- Laporte G (1988) Location-routing problems. In: Golden B, Assad A (eds) *Vehicle routing: methods and studies*. North-Holland, Amsterdam, pp 163–198
- Lee J-H, Moon I-K, Park J-H (2010) Multi-level supply chain network design with routing. *Int J Prod Res* 48(13):3957–3976
- Lopes RB, Ferreira C, Sousa Santos B (2016) A simple and effective evolutionary algorithm for the capacitated location-routing problem. *Comput Oper Res* 70:155–162
- Mavrotas G (2009) E Effective implementation of the ε -Constraint method in Multi-Objective Mathematical programming problems. *Appl Math Comput* 213:455–465
- Min H, Jayaraman V, Srivastava R (1998) Combined location-routing problems: a synthesis and future research directions. *Eur J Oper Res* 108(1):1–15



- Patia RK, Vratb P, Kumarc P (2008a) A goal programming model for paper recycling system. *Omega* 36:405–417
- Patia RK, Vratb P, Kumarc P (2008b) A goal programming model for paper recycling system. *Omega* 36:405–417
- Prins C, Prodhon C, WolerCalvo R (2006) A memetic algorithm with population management (MA/PM) for the capacitated location-routing problem. In: Gottlieb J, Raidl G (eds.) *Evolutionary computation in combinatorial optimisation (EvoCOP) Proceedings 2006*, volume 3906 of *Lecture Notes in Computer Science*, Springer, Berlin, p 183–194
- Privé J, Renaud J, Boctor F, Laporte G (2006) Solving a vehicle-routing problem arising in soft-drink distribution. *J Oper Res Soc* 57:1045–1052
- Prodhon C, Prins C (2014) A survey of recent research on location-routing problems. *Eur J Opera Res*. doi:[10.1016/j.ejor.01.005](https://doi.org/10.1016/j.ejor.01.005)
- Rath S, Gutjahr WJ (2011) A math-heuristic for the warehouse location-routing problem in disaster relief. *Comput Oper Res*. doi:[10.1016/j.cor.07.016](https://doi.org/10.1016/j.cor.07.016) (available online)
- Samanlioglu F (2013) multi-objective mathematical model for the industrial hazardous waste location-routing problem. *Eur J Oper Res* 226:332–340
- Tavakkoli-Moghaddam R, Alinaghian M, Salamat-Bakhsh A, Norouzi N (2012) A hybrid meta-heuristic algorithm for the vehicle routing problem with stochastic travel times considering the driver's satisfaction. *J Ind Eng Int* 8:4
- Tavakkoli-Moghaddam R, Forouzanfar F, Ebrahimnejad S (2013) Incorporating location, routing, and inventory decisions in a bi-objective supply chain design problem with risk-pooling. *J Ind Eng Int* 9:19
- Vidović M, Ratković B, Bjelić N, Popović D (2016) A two-echelon location-routing model for designing recycling logistics networks with profit: MILP and heuristic approach. *Expert Syst Appl* 51:34–48
- Yousefikhoshbakht M, Khorram E (2012) Solving the vehicle routing problem by a hybrid meta-heuristic algorithm. *J Ind Eng Int* 8:11
- Yu VF, Lin SW, Lee W, Ting CJ (2010) A simulated annealing heuristic for the capacitated location routing problem. *Comput Ind Eng* 58(2):288–299
- Zhalechian M, Tavakkoli-Moghaddam R, Zahiri B, Mohammadi M (2016) Sustainable design of a closed-loop location-routing-inventory supply chain network under mixed uncertainty. *Transp Res Part E Logist Transp Rev* 89:182–214

