Journal of Industrial and Systems Engineering
Vol. 11, No.2, pp. 62-84
Spring (April) 2018

A multi-objective Two-Echelon Capacitated Vehicle Routing Problem for perishable products

Mehraneh Esmaeili[1], Rashed Sahraeian[1*]

[1] *Industrial Engineering Department, College of Engineering, Shahed University, Tehran, Iran*

m_esmaili@shahed.ac.ir, sahraeian@shahed.ac.ir

Abstract

This article addresses a general tri-objective two-echelon capacitated vehicle routing problem (2E-CVRP) to minimize the total travel cost, customers waiting times and carbon dioxide emissions simultaneously in distributing perishable products. In distributing perishable products customers' satisfaction is very important this is inversely proportional to the customers waiting times. The proposed model is a mixed integer non-linear programming (MINLP). By applying some linearization methods, the MINLP model exchanged to a mixed integer linear programming (MILP). This paper uses a non-dominated sorting genetic (NSGA-II) algorithm to solve the presented mathematical model. The related results would be compared with Lp-metric results in small-sized test problems and with multi objective particle swarm optimization (MOPSO) algorithm in medium and large sized test problems. In order to evaluate the quality of the solution sets, the results of two meta-heuristic algorithms are compared based on four comparison metrics in medium sized problems. The obtained results indicate the efficiency of the NSGA-II algorithm.

Keywords: 2E-CVRP, carbon dioxide emissions, perishable products, customers waiting times, linearization, multi objective optimization

1-Introduction

The application that motivated the introduction of the 2E-CVRP is city logistics. Several studies on the 2E-CVRP cite the paper by Crainic, Ricciardi and Storchi (2009) as the one that introduced the first formal definition of a 2E-CVRP, even if the term 2E-CVRP appeared later in the literature. The problem tackled in their paper is a 2E-CVRP with time-dependent, synchronized, multi-depot, multi-product, heterogeneous fleets (on each echelon), and time windows. The authors provide an ILP formulation of the problem and design solution methods, but do not report any computational experiment (Cuda, Guastaroba and Speranza, 2013). The term 2E-CVRP was introduced in the paper of Perboli, Tadei and Vigo (2011) where a formal definition of the problem is provided. The authors propose a MILP formulation along with two families of valid inequalities and two matheuristics for the 2E-CVRP. According to review by Cuda, Guastaroba and Speranza (2013), the best performing heuristic for the 2E-CVRP is the ALNS introduced by Hemmelmayr, Cordeau and Crainic (2012), while the best exact algorithm is proposed by Baldacci et al. (2013). It is worth highlighting that some researchers compare the effectiveness of a delivery strategy implementing a 2E-CVRP with a single-

*Corresponding author
ISSN: 1735-8272,

echelon strategy adopting a CVRP. Soysal, Bloemhof-Ruwaard and Bektaş (2015) present a comprehensive MILP formulation for a time-dependent two-echelon capacitated vehicle routing problem (2E-CVRP) that accounts for vehicle type, traveled distance, vehicle speed, load, multiple time zones and emissions. The results suggest that an environmentally friendly solution is obtained from the use of a two-echelon distribution system, whereas a single-echelon distribution system provides the least-cost solution. Crainic et al. (2010) study the impact of instance parameters on the global cost. They present the results of a broad experimental study aimed at analyzing the impact on the total distribution cost of several. A branch-and-cut-and-price algorithm for the two-echelon capacitated vehicle routing problem is introduced by Santos, Mateus and da Cunha (2014). In the research by Zeng, Xu and Xu (2013) a two-phase hybrid heuristic is proposed for 2E-VRP, which is composed of a greedy randomized adaptive search procedure (GRASP) and a variable neighborhood descent (VND), called GRASP+VND in the sequel. Baldacci et al. (2013) describe a new mathematical formulation of the 2E-CVRP used to derive valid lower bounds and an exact method that decomposes the 2E-CVRP into a limited set of multi depot capacitated vehicle routing problems with side constraints. Jepsen, Spoorendonk and Ropke (2013) present an exact method for solving the symmetric two-echelon capacitated vehicle routing problem. The presented method is based on an edge flow model that is a relaxation and provides a valid lower bound. Hemmelmayr, Cordeau and Crainic (2012) propose an adaptive large neighborhood search heuristic for the Two-Echelon Vehicle Routing Problem (2E-VRP) and the Location Routing Problem (LRP). They have developed new neighborhood search operators by exploiting the structure of the two problem classes considered and have also adapted existing operators from the literature. A novel mathematical formulation of time-dependent vehicle routing problems with heterogeneous fleet, hard time widows and multiple depots, is proposed by Afshar- Nadjafi and Razmi-Farooji (2014). To deal with the traffic congestions, they also considered that the vehicles are not forced to come back to the depots, from which they were departed. To solve the problem two well-known Meta-heuristic algorithms were presented namely NSGA-II and MOSA.

Green logistics has emerged recently and has attracted some researchers attention and has a great role in supply chain management in recent years. In the traditional logistics model, only economic objective is considered, but in green logistics, societal and environmental objectives are considered in addition to the economic objective. Integrating vehicle routing problem (VRP) and green logistics developed a new field in logistics problem called Green Vehicle Routing Problem (GVRP) (Rabbani, Farrokhi-Asl and Asgarian, 2017). Lin et al. (2014) published a complete review on GVRP and provided a classification of GVRP. They categorize these problems into three branch lines including Green-VRP, Pollution Routing Problem and VRP in reverse logistics. The proposed model in this paper is a pollution routing problem since it aims at minimizing the total carbon dioxide (co_2) emissions in routing problem. The Pollution Routing Problem (PRP) aims at choosing a vehicle dispatching scheme with less pollution, in particular reduction of carbon emissions (Lin et al., 2014). Govindan et al. (2013) introduce a two-echelon location-routing problem with time-windows (2E-LRPTW) for sustainable SCN design and optimizing economic and environmental objectives in a perishable food SCN. The goal of 2E-LRPTW is to determine the number and location of facilities and to optimize the amount of products delivered to lower stages and routes at each level. It also aims to reduce costs caused by carbon footprint and greenhouse gas emissions throughout the network. The proposed method includes a novel multi-objective hybrid approach called MHPV, a hybrid of two known multi-objective algorithms: namely, multi-objective particle swarm optimization (MOPSO) and adapted multi-objective variable neighborhood search (AMOVNS). Esmaili and Sahraeian (2017) represent a two-echelon capacitated vehicle routing problem (2-ECVRP). The paper proposes a novel bi-objective model that minimizes: 1) total customers waiting time, and 2) total travel cost. A restriction on maximum allowable carbon dioxide (CO2) emissions from transport in each route is considered as environmental issue in the problem. The sensitivity analysis performed on the model reveals that less restrictive policies on carbon emissions lead to more total emissions but less total travel cost and customers waiting times.

In distribution systems customer's satisfaction is an important issue, especially for food products. In general, food products are characterized as perishable items. The quality of perishable food products decays rapidly during the delivery process. Their freshness is significantly affected by the time duration and environment temperature during the delivery. Hence, it is important that perishable foods must be delivered within allowable delivery time windows, or a penalty shall be incurred for late arrivals (Chen, Hsueh and Chang, 2009). Song and Ko (2016) developed a nonlinear mathematical model to maximize the total level of the customer satisfaction which is dependent on the freshness of delivered food products and assumed that each vehicle has a maximum allowable delivery time. Angel-Bello, Martínez-Salazar and Alvarez (2013) introduce a routing problem with multiple use of a single vehicle and service time in demand points (clients) with the aim of minimizing the sum of clients waiting time to receive service. They consider vehicle capacity and travel distance constraints which force multiple use of the vehicle in the planning horizon. The vehicle routing problem with simultaneous pickup and delivery considering customer satisfaction is presented by Fan (2011). Customer satisfaction is based on a time window at each customer location. In such a problem, the transportation requests have to be performed by vehicles, each request having to be met as early as possible. The customer's satisfaction is inversely proportional to the waiting time for the vehicle from the lower bound of the time window. Wang et al. (2016) propose a multi-objective vehicle routing problem with time windows dealing with perishability (MO-VRPTW-P). They design an effective distribution route that can minimize the total costs and maximize the freshness state of the delivered products. A two-phase heuristic algorithm based on Pareto variable neighborhood search genetic algorithm considering temporal-spatial distance (STVNS-GA) is applied to solve the problem.

The multi objective 2E-CVRP in this paper is a development of 2E-CVRP in the paper of Esmaili and Sahraeian (2017). The model in Esmaili and Sahraeian (2017) was presented by the authors of this paper. The previous research has presented a two objective model which considers the customers waiting time from satellites in second echelon. But this paper presents a tri-objective model which considers customer waiting time from depot in first echelon. The customer waiting time was'nt considered from the first echelon in pervious papers. The objectives are minimizing 1) total travel costs, 2) sum of customers waiting times and 3) total co_2 emissions. This paper also considers a maximum allowable delivery time for perishable goods that they should be delivered within that. The end customer's satisfaction has a reverse relation with his waiting time. This means that the less waiting time they have, the more satisfaction they will achieve specially for perishable products. The proposed model is a mixed integer non-linear programming (MINLP) problem. By applying some linearization methods the model exchanges to a mixed integer linear programming (MILP) problem. The MILP problem is solved by Lp-metric method in CPLEX for small size instances. Medium and large sized test problems are solved by NSGA-II algorithm. The results are compared with MOPSO algorithm through four comparison metrics.

The rest of this paper is organized as follows. Section 2 contains an optimization model to minimize the total costs, total customers waiting times and total carbon emissions and the procedure of linearization. The solution of the model and the suitable algorithm are presented in section 3. In section 4, the results obtained from the computational experiments are shown for different size of the problem. Finally, we conclude the paper in section 5 by providing several topics for future research.

2-Problem Description

Freight transportation can be broadly categorized into two classes according to the presence of one or more intermediate facilities. Direct shipping takes place when freight is delivered directly from its origin to its destination. Conversely, indirect shipping takes place when freight, or part of the freight, is moved through some intermediate facilities (e.g., cross-docks or distribution centers) before reaching its destination. Two-echelon distribution systems are a special case of multi-echelon systems where the network is composed of two echelons. In this case, after leaving its origin, freight is first delivered to an intermediate facility where storage, merging, consolidation or transshipment operations take place. The freight is then moved from the intermediate facility towards its destination. Given this framework, the flow of freight in one echelon must be coordinated with that in the other echelon. As a consequence, routing problems arising in two-echelon distribution systems cannot be merely decomposed into two sub-problems and then solved separately. Two-echelon routing problems can be classified according to the type of decisions involved (Cuda, Guastaroba and Speranza, 2013).

- Strategic planning decisions: they include decisions concerning the infrastructure of the network, typically the number and the location of the facilities.
- Tactical planning decisions: they include the routing of freight through the network and the allocation of customers to the intermediate facilities.

Cuda, Guastaroba and Speranza (2013) refer to the Two-Echelon Vehicle Routing Problems (2E-VRPs) when the problem definition involves only tactical planning decisions, and the routing is present at both echelons. In a 2E-VRP the set of depot and the set of satellites to use is given, and no cost is associated with the use of any depot and any satellite.

The two-echelon capacitated vehicle routing problem (2E-CVRP) is a distribution system where intermediate facilities, known as satellites are capacitated (Perboli, Tadei and Vigo, 2011). Satellites capacities are limited according to the number of second level vehicles in it. The 2E-CVRP is an NP-hard problem. Figure 1 shows a feasible solution to the 2E-CVRP.

The current paper proposes a tri-objective two-echelon capacitated VRP. The goal is to simultaneously minimize the total travel costs, the customers waiting times and total co_2 emissions for perishable product delivery. In perishable goods distribution the customer satisfaction is inversely proportional to customer waiting time. If the customer has less waiting time, it means that the product has passed less time in truck; so, it is fresher and more customer satisfaction would be achieved. So minimizing customers waiting times can lead to customers' satisfaction. This research counts the customers waiting time from depot in first echelon which wasn't considered in previous studies.

Because of the nature of perishable products they should be delivered within a time window. They would be decayed after the maximum allowable time. Thus a maximum allowable delivery time is considered for products delivery, inspired from Song and Ko (2016). In order to measure the environmental impact of the transportation network, GH_1 and GH_2 coefficients are considered as CO_2 produced by first level and second level vehicles in each unit of distance, respectively. Considering these coefficients is inspired from Mirzapour Al-e-hashem and Rekik (2014).

The two echelon base model is adopted from Perboli, Tadei and Vigo (2011) and this paper is a development of author's pervious research in Esmaili and Sahraeian (2017).

To formulate the mathematical model, the assumptions are as follows:

1. Vehicles in the same level have the same capacity and speed.

2. Fixed costs of the vehicles are not considered, since they are available in fixed numbers.

3. Each satellite receives freight from one or more 1st level vehicles, but each customer receives its freight from one of the 2nd level vehicles.

4. The satellites are capacitated and each satellite is supposed to have its own capacity, usually expressed in terms of maximum number of secondary vehicles in it.

5. There is just one perishable product type which has to be distributed.

6. The traveling cost of each unit of distance is equal to 1 ($\acute{c}_{ij} = 1$).

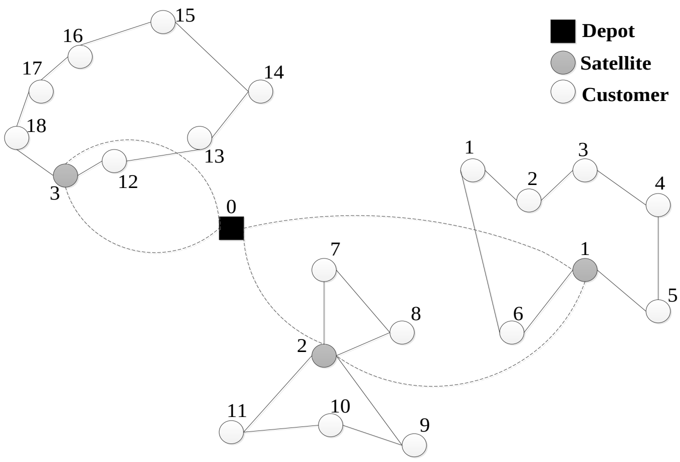

Fig 1. A feasible solution of the 2E-CVRP

The proposed mathematical model is as follows:

2-1-Sets

V_O	Depot
V_S	Set of satellites and $k \in V_S$
V_C	Set of customers and $j \in V_C$

2-2-Parameters

n_s	Number of satellites
n_c	Number of customers
m_1	Number of the 1st-level vehicles
m_2	Number of the 2nd-level vehicles
m_{s_k}	Capacity of the satellite k
K^1	Capacity of the 1st level vehicles
K^2	Capacity of the 2nd level vehicles
d_i	Demand required by customer i
C_{ij}	Length of the arc (i, j)
S_k	Cost for loading/unloading operations of a unit of freight in satellite k
GH_1	Carbon emissions in each distance unit for first level vehicle (kg/km)
GH_2	Carbon emissions in each distance unit for second level vehicle (kg/km)
v_1	Speed of 1st level vehicles (km/h)
v_2	Speed of 2nd level vehicles (km/h)
T_{max}	Maximum allowable travel time for each perishable product
st_j	The service time in node j, $st_o = 0$

2-3-Decision Variables

Q^1_{ij}	Flow passing through the 1st-level arc (i, j)
Q^2_{ijk}	Flow passing through the 2st-level arc (i, j) and coming from satellite k
x_{ij}	Number of 1st-level vehicles using the 1st-level arc (i, j)
y^k_{ij}	Boolean variable equal to 1 if the 2nd-level arc (i, j) is used by the 2nd-level vehicle starting from satellite k
y_{ij}	Boolean variable equal to 1 if the 1st-level arc (i, j) is used by the 1st-level vehicle
D_k	The freight passing through satellite k
t_j	Total waiting time for customer j
t^1_k	Elapsed time to get to satellite k in first level, $t^1_o = 0$
t_{ij}	The time spent in second level arc (i, j)
t^1_{ij}	The time spent in first level arc (i, j)

2-4-Mathematical formulation

We now present the formulation, starting with the objective functions:

$$\min Z_1 = \sum_{i,j \in V_O \cup V_S, i \neq j} C_{ij} \acute{c}_{ij} x_{ij} + \sum_{k \in V_S} \sum_{i,j \in V_S \cup V_C, i \neq j} C_{ij} \acute{c}_{ij} y^k_{ij} + \sum_{k \in V_S} S_k D_k \tag{1}$$

$$\min Z_2 = \sum_{j \in V_C} t_j \tag{2}$$

$$\min Z_3 = \sum_{i,j \in V_S \cup V_O, i \neq j} GH_1 C_{ij} y_{ij} + \sum_{k \in V_S} \sum_{i,j \in V_S \cup V_C, i \neq j} GH_2 C_{ij} y^k_{ij} \tag{3}$$

The objective function (1) minimizes sum of the traveling and handling operations costs. It comprises three parts. The two first parts are total travel costs in first and second echelon, respectively. The third part is total handling operations costs in satellites. The second objective function (2) is minimizing the total customers waiting times. The third objective (3) minimizes the total CO_2 emissions in both first and second level.

$$\sum_{k \in V_S} x_{ik} \leq m_1 \qquad \forall i \in V_S \cup V_O \tag{4}$$

$$\sum_{k\in V_S}\sum_{j\in V_C} y_{kj}^k \le m_2 \tag{5}$$

$$\sum_{j\in V_C} y_{kj}^k \le m_{s_k} \qquad \forall k \in V_S \tag{6}$$

Constraints (4)–(6) are related to the number of vehicles. The number of first and second level vehicles is considered in constraints (4) and (5), respectively. Constraints (6) ensure the capacity of the satellites.

$$\sum_{j\in V_S\cup V_O, k\neq j} x_{jk} = \sum_{i\in V_S\cup V_O, i\neq k} x_{ki} \qquad \forall k \in V_S \cup V_O \tag{7}$$

$$\sum_{i\in V_S\cup V_O, i\neq j} y_{ij} = \sum_{i\in V_S\cup V_O, i\neq j} y_{ji} \qquad \forall j \in V_S \cup V_O \tag{8}$$

$$\sum_{j\in V_C} y_{kj}^k = \sum_{j\in V_C} y_{jk}^k \qquad \forall k \in V_S \tag{9}$$

$$\sum_{k\in V_S}\sum_{i\in V_C, i\neq j} y_{ij}^k + \sum_{k\in V_S} y_{kj}^k = 1 \qquad \forall j \in V_C \tag{10}$$

$$\sum_{k\in V_S}\sum_{i\in V_S\cup V_C, i\neq j} y_{ij}^k = \sum_{k\in V_S}\sum_{i\in V_S\cup V_C, i\neq j} y_{ji}^k = 1 \qquad \forall j \in V_C \tag{11}$$

$$\sum_{i\in V_S\cup V_C} y_{ij}^k = \sum_{l\in V_S\cup V_C} y_{jl}^k \qquad \forall j \in V_C, \forall k \in V_S \tag{12}$$

$$y_{kj}^k \le \sum_{l\in V_S\cup V_O} x_{kl} \qquad \forall k \in V_S, \forall j \in V_C \tag{13}$$

$$x_{ij} \le M y_{ij} \qquad \forall i,j \in V_S \cup V_O, i \neq j \tag{14}$$

$$x_{ij} \ge y_{ij} \qquad \forall i,j \in V_S \cup V_O, i \neq j \tag{15}$$

Relations (7)-(15) are routing constraints. (7) and (8) equalities are first level routing constraints while second level routing constraints are (9)-(13). Constraints (7)-(9) and (11)-(12) indicate that in a node both in first and second level, the number of input paths is equal to the number of output paths. Inequalities in (10) ensure the visiting of each customer. Constraints (13) relate the two echelons to each other. These constraints allow a 2nd-level route to start from satellite k just once a 1st-level route has served it. The relation between x_{ij} and y_{ij} is peresented by (14) and (15) constraints which express that if some primary vehicles are used in a first level path (x_{ij}>0), that path should be selected (y_{ij}=1) and should not be selected, otherwise.

$$\sum_{i\in V_S\cup V_O, i\neq j} Q_{ij}^1 - \sum_{i\in V_S\cup V_O, i\neq j} Q_{ji}^1 = \begin{cases} D_j & j \text{ is not the depot} \\ \sum_{i\in V_C} -d_i & otherwise \end{cases} \qquad \forall j \in V_S \cup V_O \tag{16}$$

$$\sum_{i\in V_S\cup V_C, i\neq j} Q_{ijk}^2 - \sum_{i\in V_S\cup V_C, i\neq j} Q_{jik}^2 = \begin{cases} \sum_{i\in V_S\cup V_C, i\neq j} y_{ijk} d_j & j \text{ is not a satellite} \\ -D_j & \end{cases} \tag{17}$$

$$Q_{ij}^1 \le K^1 x_{ij} \qquad \forall i,j \in V_S \cup V_O, i \neq j \tag{18}$$

$$Q_{ijk}^2 \le K^2 y_{ij}^k \qquad \forall i,j \in V_S \cup V_C, i \neq j, k \in V_S \tag{19}$$

$$\sum_{i\in V_S} Q_{iv_o}^1 = 0 \tag{20}$$

$$\sum_{j\in V_C} Q_{jkk}^2 = 0 \qquad \forall k \in V_S \tag{21}$$

$$D_k = \sum_{j \in V_C} \sum_{i \in V_S \cup V_C, i \neq j} d_j y_{ijk} \qquad \forall k \in V_S \qquad (22)$$

Constraints (16)-(22) are flow constraints. Equations in (16) assure that in first echelon the demand of node j if it is a satellite is equal to its demand and if it is depot is equal to the minus of total customers' demands. Equation (17) indicates that in second echelon the demand of node j if it is a customer is equal to his demand and if it is a satellite is equal to the minus of its demand. In fact, each node receives an amount of flow equal to its demand to prevent the presence of sub tours.

The vehicle capacity constraints are formulated in (18) and (19), for the 1st-level and the 2nd-level, respectively. Constraints (20) and (21) do not allow residual flows in the routes, making the returning flow of each route to the depot (1st-level) and to each satellite (2nd-level) equal to 0.
Equations (22) count the total demand of satellite k.

$$t_{ij}^1 \times v_1 = C_{ij} \qquad i,j \in V_S \cup V_O \qquad (23)$$
$$t_{ij} \times v_2 = C_{ij} \qquad i,j \in V_S \cup V_C \qquad (24)$$
$$t_k^1 = \max_{i \in V_O \cup V_S, i \neq k} \{(t_i^1 + st_i + t_{ik}^1) y_{ik}\} \qquad \forall k \in V_S \qquad (25)$$
$$t_j = \sum_{k \in V_S} (t_k^1 + st_k + t_{kj}) y_{kj}^k + \sum_{k \in V_S} \sum_{i \in V_C, i \neq j} (t_i + st_i + t_{ij}) y_{ij}^k \qquad \forall j \in V_C \qquad (26)$$
$$t_j \leq T_{max} \qquad \forall j \in V_C \qquad (27)$$

Time constraints are formulated in (23)-(27). Equations (23) and (24) calculate the required time to pass arc (i,j) in first and second level, respectively. Constraints (25) show the arrival time to satellite k. It is the summation of arrival time to pervious node, service time in that node and required time to pass the arc (i,k) in first level. The arrival time to customer j is formulated in constraints (26). It consists of two parts. First part is used when the customer j is visited directly after satellite k and second part usage is when the customer j is visited after customer i. Inequalities in (27) ensure that the arrival time to customer j don't exceed the maximum allowable perishable product delivery time.

$$y_{ij}^k \in \{0,1\} \qquad i,j \in V_S \cup V_C\, , k \in V_S \qquad (28)$$
$$y_{ij} \in \{0,1\} \qquad i,j \in V_S \cup V_O \qquad (29)$$
$$x_{ij} \in Z^+ \qquad i,j \in V_S \cup V_O \qquad (30)$$
$$Q_{ij}^1 \geq 0 \qquad i,j \in V_S \cup V_O \qquad (31)$$
$$Q_{ijk}^2 \geq 0 \qquad i,j \in V_S \cup V_C, k \in V_S \qquad (32)$$
$$t_k^1 \geq 0 \qquad k \in V_S, t_O^1 = 0 \qquad (33)$$
$$t_{ij}^1 \geq 0 \qquad i,j \in V_S \cup V_O \qquad (34)$$
$$t_{ij} \geq 0 \qquad i,j \in V_S \cup V_C \qquad (35)$$
$$t_j \geq 0 \qquad j \in V_C \qquad (36)$$

Constraints (28)–(36) represent the binary and non-negativity restrictions imposed on the decision variables. Constraints (28) and (29) show the binary variables. Constraint (30) shows positive integer variables. Finally, constraints (31)-(36) show the positive continuous variables.

2-5-Linearization

The proposed model in section 3.4 is a nonlinear mathematical model. Constraints (25) and (26) are the non-linear constraints because they both have production of continues and binary variables. Besides, Constraints (26) are maximizing expressions. Linearization procedures for these constraints are as follows:

According to Glover and Woolsey (1974) for linearizing the product of a binary and a continuous variable suppose your expression is $Z = x_1 \times x_2$ where x_2 is a continuous variable and x_1 is a binary variable. Now if $x_1 = 1$ then Z has to be equal to continuous variable; otherwise Z is zero. For linearizing this expression Glover and Woolsey added the inequalities (37)-(39).

$$Z \leq x_2 \tag{37}$$

$$Z \leq M \times x_1 \tag{38}$$

$$Z \geq x_2 - M(1 - x_1) \tag{39}$$

Therefore, in order to linearizing constraints (25) and (26) the inequalities sets (40)-(41) and (42)-(45) will be added, respectively.

$$t_i^1 \times y_{ik} = Z_1(i,k) \tag{40}$$

$$Z_1(i,k) \leq t_i^1 \qquad Z_1(i,k) \leq M \times y_{ik} \qquad Z_1(i,k) \geq t_i^1 - M(1 - y_{ik})$$

$$t_{ik}^1 \times y_{ik} = Z_2(i,k) \tag{41}$$

$$Z_2(i,k) \leq t_{ik}^1 \qquad Z_2(i,k) \leq M \times y_{ik} \qquad Z_2(i,k) \geq t_{ik}^1 - M(1 - y_{ik})$$

$$t_k^1 \times y_{kj}^k = Z_3(k,j) \tag{42}$$

$$Z_3(k,j) \leq t_k^1 \qquad Z_3(k,j) \leq M \times y_{kj}^k \qquad Z_3(k,j) \geq t_k^1 - M \times (1 - y_{kj}^k)$$

$$t_{kj}^1 \times y_{kj}^k = Z_4(k,j) \tag{43}$$

$$Z_4(k,j) \leq t_{kj}^1 \qquad Z_4(k,j) \leq M \times y_{kj}^k \qquad Z_4(k,j) \geq t_{kj}^1 - M \times (1 - y_{kj}^k)$$

$$t_i \times y_{ij}^k = Z_5(i,j,k) \tag{44}$$

$$Z_5(i,j,k) \leq t_i \qquad Z_5(i,j,k) \leq M \times y_{ij}^k \qquad Z_5(i,j,k) \geq t_i - M \times (1 - y_{ij}^k)$$

$$t_{ij} \times y_{ij}^k = Z_6(i,j,k) \tag{45}$$

$$Z_6(i,j,k) \leq t_{ij} \qquad Z_6(i,j,k) \leq M \times y_{ij}^k \qquad Z_6(i,j,k) \geq t_{ij} - M \times (1 - y_{ij}^k)$$

By adding these inequalities sets, constraints (25) and (26) modify to constraints (46) and (47), respectively.

$$t_k^1 = \max_{i \in V_O \cup V_S, i \neq k} \{ Z_1(i,k) + st(i) \times y_{ik} + Z_2(i,k) \} \qquad \forall k \in V_S \tag{46}$$

$$t_j = \sum_{k \in V_S} (Z_3(k,j) + st(k) \times y_{kj}^k + Z_4(k,j)) + \sum_{k \in V_S} \sum_{i \in V_C, i \neq j} (Z_5(i,j,k) + st_i \times y_{ij}^k + Z_6(i,j,k)) \qquad \forall j \in V_C \tag{47}$$

$$Z_l(i,k) \geq 0 \qquad k \in V_S, i \in V_O \cup V_S, l = 1,2 \tag{48}$$

$$Z_m(k,j) \geq 0 \qquad k \in V_S, j \in V_C, m = 3,4 \tag{49}$$

$$Z_h(i,j,k) \geq 0 \qquad k \in V_S, i \in V_C, j \in V_C, h = 5,6 \tag{50}$$

The inequalities in (46) by having maximization expressions are not linear. For linearizing these constraints according to operational research methods, inequalities in (52) are added.

$$X = \max_i \{ a(i) \} \tag{51}$$

$$X \geq a(i) \qquad \forall i \tag{52}$$

So, the maximization constraints are linearized.
Constraints (46) will be replaced by (53) and (54) inequalities to be linearized.

$$Z_1(i,k) + st(i) \times y_{ik} + Z_2(i,k) = time(i,k) \qquad \forall i \in V_O \cup V_S, \forall k \in V_S \tag{53}$$

$$t_k^1 \geq time(i,k) \qquad \forall i \in V_O \cup V_S, \forall k \in V_S \tag{54}$$

Therefore, after all these procedures the proposed mixed integer nonlinear programming (MINLP) model is linearized and exchange to a mixed integer linear programming (MILP) model. The linearized model will be solved in next section.

3-Solution method

There are two general approaches for optimizing multi objective problems: Classic approaches and evolutionary techniques. Classic approaches include decomposition methods such as sum of weighted additive (SAW), ε-constraint, Lp-metric, goal programming and goal attainment methods. These approach exchange the multi objective problem to single objective problem. Each run of the model can find a Pareto solution. So to find a Pareto front, the problem must be run several times. These approaches can be implemented by the aid of business solvers and usually they are not appropriate for large sized problems. But the evolutionary techniques are direct methods for problem solving. These techniques can find the Pareto front at once. They are fast and usually appropriate for large sized problems, such as multi objective meta-heuristics.

In this paper to solve the problem by decomposition methods, the Lp-metric method is considered and to solve the problem by evolutionary techniques, NSGA-II and MOPSO are implemented.

3-1-Lp-metric method

Lp metric method is a simple method for forming a combined objective function. The purpose of Lp metric method is minimizing deviations of the existing objective functions from an ideal solution. Therefor the overall objective function will be as follows (Mahmoodjanloo, Esmaili and Hajiaghaei-Keshteli, 2016):

$$MinT = \left(\sum_{i=1}^{n} w_i (\frac{f_i - f_i^*}{f_i^*})^r\right)^{1/r} \tag{55}$$

f_i: The amount of objective function i

f_i^*: The amount of ideal objective function i

w_i: Importance coefficient of objective function i

r: Parameter to emphasize the deviations

3-2-NSGA-II for solving the problem

3-2-1-NSGA-II algorithm

Genetic algorithm proposed by John Holland (1975) is one of the most popular and applicable evolutionary meta-heuristic to solve problems which are hard to find exact and optimal solutions. NSGA-II is an extension for the genetic algorithm and is used for solving the multi objective problems and is proposed by Deb et al. (2002). Algorithm starts with the initialization step. In this step, initial population is generated according to the section mentioned previously. After this step obtained solutions are improved each iteration by means of crossover and mutation operators.

The manner in which fitness function is calculated is critical and important factor in evolutionary algorithms (Rabbani, Farrokhi-Asl and Asgarian, 2017). A flowchart of the algorithm is depicted in figure 2.

3-2-2-Solution representation in NSGA-II

The manner in which the solutions of the problem are coded has a significant impact on the quality of the solutions and computational time. The initial structure of the solutions in both algorithms is described.

The initial structure of the solutions in NSGA-II algorithm is made up from two strings of permutations. These structures in the context of NSGA-II are called chromosomes. The first string shows all routes between satellites and customers. It’s related to second level routes and vehicles.

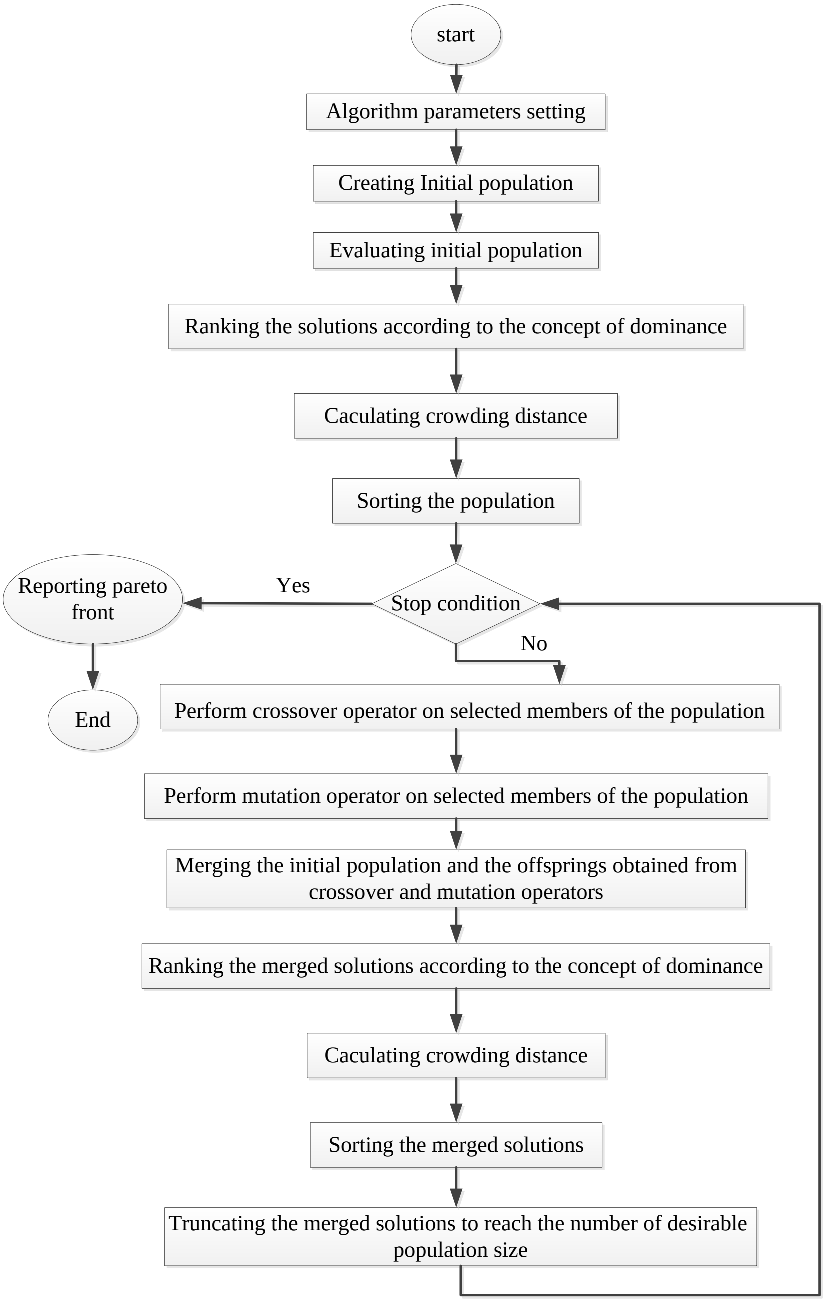

Fig 2. Flow chart of NSGA-II algorithm

This string consists of $n_c + m_2 - 1$ permutation numbers. The numbers 1 to n_c, represent the customers. The numbers between n_c and $n_c + n_s$, show satellite's delimiters. They show also that the numbers bigger than n_c are vehicle's delimiters. For example, by considering 4 customers, 2 satellites and 3 secondary vehicles a feasible permutation would be as figure 3.

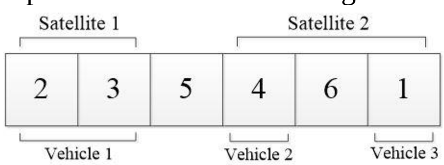

Fig 3. Second level string in NSGA-II

In this permutation a secondary vehicle is allocated to satellite1 and two ones are allocated to satellite2. The attained routes are 1-2-3-1, 2-4-2 and 2-1-2.

Note that cumulative demands of customers in each second level route must not exceed the capacity of secondary vehicle. Also, the maximum allowable delivery time constraints and the satellite's capacity constraint must be considered. We use penalty in all objective functions for trespassing from these limitation in each secondary route.

Second string depicts the routes between the depot and satellites. This string contains $(n_s * m_1) + m_1 - 1$ numbers. Since it is allowable to visit a satellite for several times, the number of that satellite should be repeated as many as the primary vehicles and $m_1 - 1$ zeros should be added to the set as primary vehicles delimiters. Then a random permutation of the numbers should be created. For example by considering 2 satellites and 2 primary vehicles a feasible permutation is as figure 4. The created routes are 0-1-2-0 and 0-2-1-0.

It must be attended that cumulative demands of satellites in each first level route must not exceed the capacity of primary vehicle. Also, the maximum allowable delivery time constraints must be considered. We use penalty in all objective functions for trespassing from these limitation in each primary route.

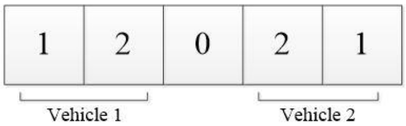

Fig 4. First level string in NSGA-II

3-2-3-Crossover operator

The performance of NSGA-II algorithm is highly dependent on crossover and mutation operators. By these operators, we can search in the solution area and explore new solutions and exploit good solutions. In genetic algorithm literature, there are vary crossover and mutation operators that can be used according to type of problem (Rabbani, Farrokhi-Asl and Asgarian, 2017). In this paper the permutation crossover operator is implemented on the population. First, we select parents to perform crossover operation by binary tournament selection operator. After selection of parents, we produce one integer random number (c) between 1 and the number of chromosome's genes. The first part of the parents is from 1 to c genes and the others are second part of the parents. The first offspring is constructed by first part of the first parent and second part of the second parent. There may be some repetitive genes in a offspring. The repetitious genes of first offspring should be replaced by repetitious genes of second offspring and vice versa. The crossover operator is applied on both first and second string.

3-2-4-Mutation operator

First a parent is selected by binary tournament selection operator. After that, we produce one integer random number between 1 and 3. In case 1, swap mutation is applied on both first and second string. In case 2 and 3, reversion and insertion mutation is applied on the strings, respectively. Swap mutation selects two genes in chromosome randomly and swaps them to generate new solutions. In reverse mutation, two genes are selected randomly and the genes position between two selected genes is reversed. In insertion, two genes are selected randomly and the second one is inserted after the first one. The mutation operator is applied on both first and second string.

3-3-MOPSO for solving the problem

3-3-1-MOPSO Algorithm

MOPSO firstly introduced by Eberhart & Kennedy (1995), is a population-based algorithm that works on the basis of social behavior of animals. Similar to chromosomes in GA, particles in this algorithm are generated and evaluated in order to find a global optimal solution in a D-dimensional space. The particles use three important features, namely velocity, position, and the best personal position, to depart from one point to another in the solution space. These features have essential effect on the quality of the solutions obtained using MOPSO. The position of each particle in the solution space is determined by an objective function value while the velocity of each particle is defined as the distance that the particle should pass from one position to another. The best position that a particle visits during its movement is called its best personal position. The mathematical expressions of the velocity and the position of each particle are obtained using the following equations (Maghsoudlou et al., 2016):

$$V_i^{t+1} = w \times V_i^t + C_1 \times r_1 \times \left(Pbest_i^t - X_i^t\right) + C_2 \times r_2 \times (gbest_i^t - X_i^t) \quad (56)$$
$$X_i^{t+1} = X_i^t + V_i^{t+1} \quad (57)$$

The following notations are used in equations (56) and (57). These equations are used to update the particles velocity and position in each iteration.

- V_i^{t+1} and V_i^t are the velocity of the i^{th} particle in iterations t and $t+1$
- X_i^{t+1} and X_i^t are the position of the i^{th} particle in iterations t and $t+1$
- $Pbest_i^t$ is the best position of i^{th} particle in iteration t
- $gbest_i^t$ is the best position among all the particles in iteration
- C_1 and C_2 are positive constant values
- r_1 and r_2 are random values between 0 and 1
- w is the inertia weight

3-3-2-Solution Representation in MOPSO

Here the solution representation is like NSGA-II. The difference is in the way of number generation. In NSGA-II the strings are a permutation of the numbers, but in MOPSO first some real numbers are generated randomly. One integer number is assigned for each real number consecutively. After this, the real numbers are sorted in ascending order and the assigned numbers are moved with corresponding real numbers. The result of this work is a random permutation of integer numbers. The first string consists of $n_c + m_2 - 1$ real numbers and its permutation depicted in table 1.

Table 1: Second level string in MOPSO

Integer numbers	1	2	3	4	5	6
Real numbers	0.95	0.16	0.29	0.72	0.41	0.84
Rank of real numbers	6	1	2	4	3	5
Permutation	2	3	5	4	6	1

The assignment to satellites and secondary vehicles are similar to NSGA-II solution representation. The second string is consisting of $(n_s * m_1) + m_1 - 1$ real numbers. They would be sorted in ascending order. Then the first m_1 ranks are 1, the next m_1 ranks are 2 and so on. The remained ranks are 0. The permutation is shown in table 2.

Table 2. First level string in MOPSO

Integer numbers	1	2	3	4	5
Real numbers	0.07	0.69	0.51	0.22	0.35
Rank of real numbers	1	5	4	2	3
Sorted ranks	1	4	5	3	2
Permutation	1	2	0	2	1

The permutation states that the first primary vehicle serves satellites 1 and 2 while the second primary vehicle serves satellites 2 and 1. The first level routes are 0-1-2-0 and 0-2-1-0.

4-Experimental results

In this section the results of Lp-metric method in CPLEX software on small sized instances are reported. The Lp-metric method on model is coded in IBM ILOG CPLEX 12.6.0.0. The VRP and also 2E-VRP problems are NP-hard. For solving the medium and large sized problems, the NSGA-II meta-heuristic is implemented on the model. The results of proposed technique are compared based on four comparison metrics in medium and large sized problems with MOPSO algorithm. The algorithms are coded in MATLAB R2015a and executed on Intel Core i7 CPU 2.00 GHz personal computer with 8 GB RAM.

4-1-NSGA-II Parameters setting

In this study, Taguchi method is utilized because it is based on the design of experiment (DOE). The purpose of the design of experiments is performing a set of tests that make meaningful changes in input variables to evaluate the impact of response variables. The response variable is obtained at the end of each experiment. A number of experiments are obtained considering how many factors and levels are defined. For NSGA-II algorithm four factors and three levels for each factor are considered. The factors are Crossover rate (pc), Mutation rate (pm), Maximum Iterations (MaxIt) and population (npop). They are shown in table 3. By considering these factors and levels L9 design is recommended for NSGA-II. The outputs are normalized using formula (56). The response variable contains four measures of NPS, DM, MID and SNS. The weighted sum of normalized measures (RPD) represents the response variable. The NPS, DM, MID and SNS weights are 2, 3, 4 and 3, respectively.

Table 3. Considered levels for NSGA-II parameters

levels	pc	pm	MaxIt	npop
low	0.6	0.1	100	50
medium	0.8	0.2	300	70
high	0.9	0.4	500	90

$$RPD = \frac{|each\ solution - best\ solution|}{best\ solution} \times 100 \tag{56}$$

- Number of Pareto Solutions (NPS): The number of non-dominated solutions that the algorithm can find. The bigger one is better one (Rabbani, Farrokhi-Asl and Asgarian, 2017).
- Diversification Metric (DM): measures the spread of the non-dominated solution set and is calculated as follows:

$$DM = \sqrt{\sum_{i=1}^{n} \max\left(\|x'_i - x'_j\|\right)} \qquad j \in 1,2,\dots,n \tag{57}$$

Where $\|x'_i - x'_j\|$ is the Euclidean distance between the non-dominated solutions x'_i and x'_j. The algorithm with a higher mean distance (DM) has a better capability (Govindan et al, 2013).

- Mean Ideal Distance (MID): calculates the mean distance between the non-dominated set and ideal point $\vec{f}_{Ideal}$. The equation of MID is computed as: $MID = \sum_{i=1}^{n} c_i / n$, where n is the number of non-dominated solutions. $\vec{f}_{Ideal}$=(0,0,0) and $c_i = \|\vec{f}_i - \vec{f}_{Ideal}\|$.

The algorithm with a lower value of MID has better performance (Govindan et al, 2013).

- Spread of Non-dominance Solution (SNS): pigeonholed as a diversity measure, it evaluates standard deviation of ideal solution from Pareto solutions and is calculated as follows:

$$SNS = \sqrt{\frac{\sum_{i=1}^{n} (MID - c_i)^2}{n-1}} \tag{58}$$

The higher value of SNS brings the better solution quality (Govindan et al, 2013). The computations are executed in 2013 Minitab 16.2.4. on the instance E-n13-k4-62. In order to interpret results, two criteria of the mean of means and signal to noise ratio (SN) are defined.

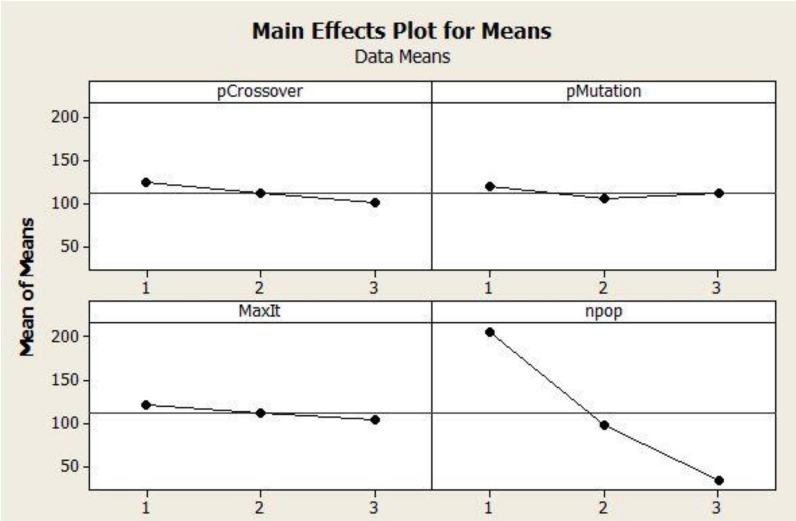

Fig 5. Mean of means of instance E-n13-k4-17

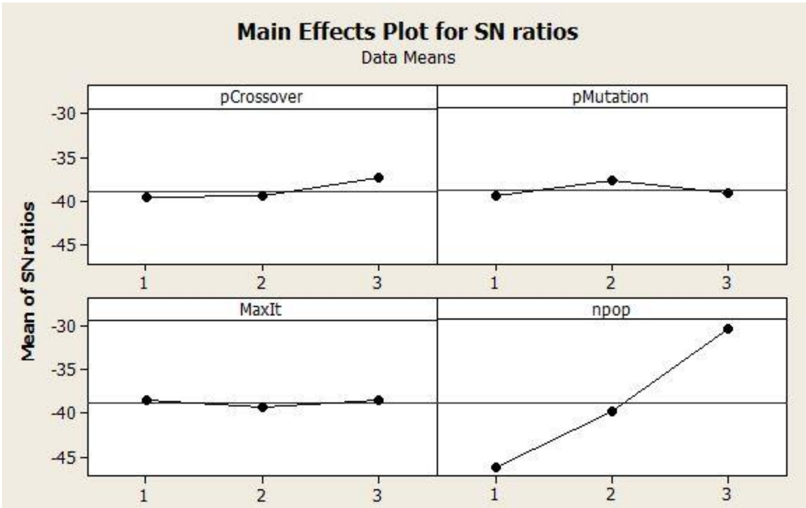

Fig 6. SN ratio of instance E-n13-k4-17

The lower values of the mean of means and the higher SN ratio show the better results. Each experiment is performed five times and the metrics in an experiment are expressed as the average of five iteration. The tuned parameters are population; crossover rate, mutation rate, and iteration which based on figure 5 and 6 are set to 90, 0.9, 0.2 and 500, respectively.

4-2-Small sized problems

The 2E-VRP is a Np-hard problem. By enhancing the size of the problem, CPU time will increase exponentially. So, the CPLEX solver is economic just in small sized problems. We consider three small sized problems. In these problems respectively 10, 11 and 12 nodes of E-n13-k4-62 instance in (Christofides and Eilon, 1969) data sets which are presented on OR-Library website are considered.

In these problems we have 2 satellites and 1 depot. Solving the larger instances would take more than one hour time. The two methods results are reported in table 4. The $f_i^{best}$is the best amount of ith objective function in pareto front. NSGA-II and CPLEX column values in table 4 are the average of nine runs of the NSGA-II algorithm and the average of nine different weight combinations of objective functions in Lp-metric method, respectively.

Table 4. Results of CPLEX and NSGA-II algorithm on small sized problems

row	nodes	n_s	n_c	f_1^{best}		f_2^{best}		f_3^{best}		CPU time(s)	
				NSGA-II	CPLEX	NSGA-II	CPLEX	NSGA-II	CPLEX	NSGA-II	CPLEX
1	10	2	7	265.66	247.97	14.10	16.55	149.2	129.11	460.04	8.53
2	11	2	8	282.37	274	15.98	18.35	155.68	138	448.25	139.50
3	12	2	9	286.38	306	17.75	21.27	155.33	167.64	465.18	524.63

One can observe that NSGA-II is capable to find near optimal solutions and in two first case can even find a better f_2^{best}.

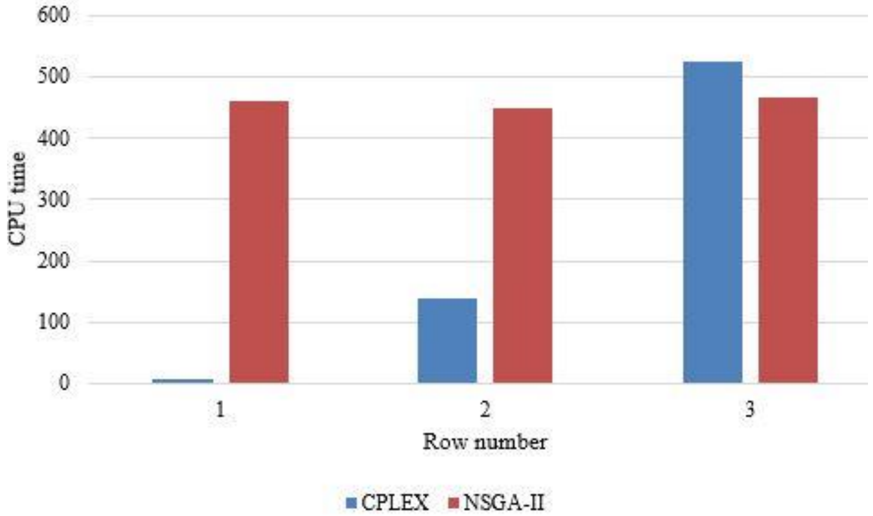

Fig 7. Comparing CPU times for solving small sized instances by NSGA-II and CPLEX

The run times in figure 7 show that time increment in NSGA-II is lower than CPLEX. A feasible solution of instance 3 of table 4 achieved by CPLEX is depicted in figure 8.

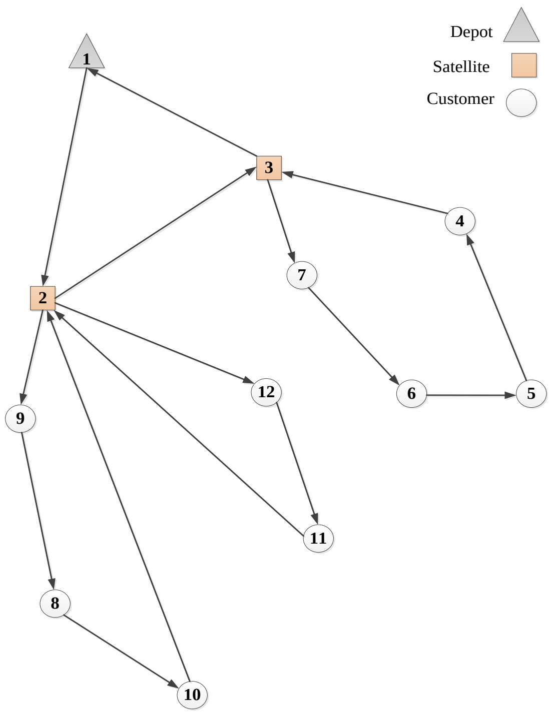

Fig 8. A feasible solution of third small sized instance

4-3-Medium sized problems

We consider instances cover up to 51 nodes (1 depot and 50 customers and 5 satellites) and which are grouped in four sets as medium sized problems. The first three sets have been built from the existing instances for VRP by Christofides and Eilon denoted as E-n13-k4, E-n22-k4, E-n33-k4 and E-n51-k5 (1969), while the fourth set is taken from Crainic et al. (2010), comprises randomly generated instances simulating different geographical distributions, including customer's distribution in urban and regional areas. All the instance sets can be downloaded from OR-Library website (Beasley, 1990). These instances are used in Perboli, Tadei and Vigo (2011).

These instances and the results of NSGA-II and MOPSO algorithms are presented in table 5. In order to parameter setting in MOPSO algorithm some assumptions are considered. Inertia weight (w) is 0.6, the cognitive learning factor (c_1) and the social learning factor (c_2) are 1 and 2, the mutation rate is 0.1, quantity of repository members (nrep) is 90 and the maximum iteration (MaxIt) is 500. As it is seen in this table, MOPSO cannot find a feasible solution for instances 13, 14, 16, 17 and 21 after five algorithm repetitions for each instance while NSGA-II has found a feasible solution for all instances.

Figures 9, 10 and 11 depict the comparison of three objective functions; respectively in the cases both algorithms presented feasible solutions. It can be seen that NSGA-II algorithm represented much better results for all objective functions.

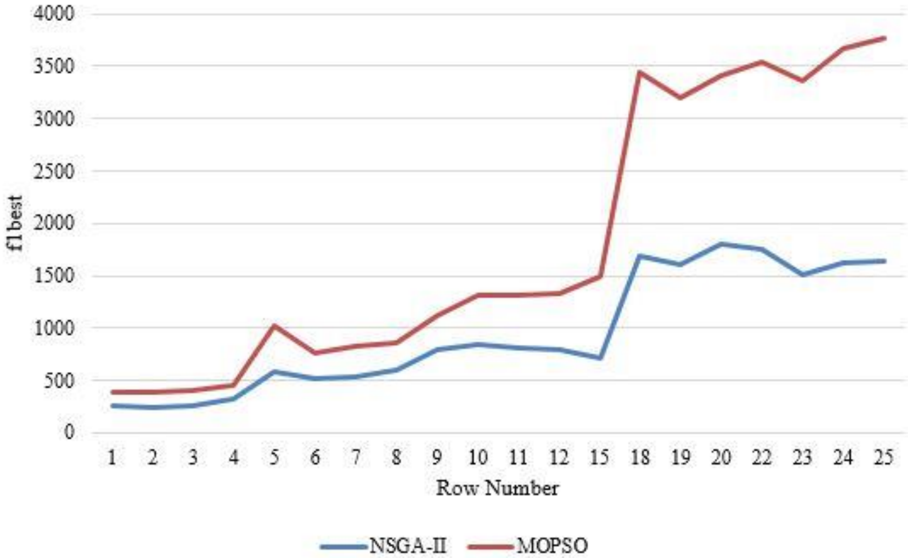

Fig 9. Comparison of f_1^{best} in NSGA-II and MOPSO algorithms

It shows the superiority of the NSGA-II versus MOPSO in such problems. In these three figures it is observable that by increasing the instance size the difference of two algorithms in solution presentation will increase.

Table 5. Results of NSGA-II and MOPSO

row	Instance Name	NSGA-II			MOPSO		
		f_1^{best}	f_2^{best}	f_3^{best}	f_1^{best}	f_2^{best}	f_3^{best}
1	E-n13-k4-17	255.42	19.57	131.04	388.22	24.49	207.84
2	E-n13-k4-39	250.38	20.6	117.28	396.3	26.56	194.16
3	E-n13-k4-48	265.36	21.45	132.72	408.26	25.9	208.08
4	E-n13-k4-62	321.86	24.22	167.6	454.5	31.2	231.2
5	E-n22-k4-s9-19	589.88	56.01	280.6	1020.18	88.78	488.69
6	E-n22-k4-s10-14	518.63	54.09	218.38	761.29	71.04	335.4
7	E-n22-k4-s13-17	541.64	56.95	250.93	832.79	73.84	412.28
8	E-n22-k4-s19-21	600.87	59.61	248.02	862.62	85.66	372.91
9	E-n33-k4-s2-13	788.36	117.03	367.97	1124.62	178.46	602.6
10	E-n33-k4-s7-25	835.72	119.06	417.98	1313.26	189.29	644.01
11	E-n33-k4-s16-24	808	113.27	353.91	1320.45	176.27	578.31
12	E-n33-k4-s22-26	788.4	114.14	362.40	1333.82	187.48	628.96
13	E-n51-k5-13-44	936.47	179.87	416.79	-	-	-
14	E-n51-k5-40-42	1018.37	194.5	465.81	-	-	-
15	E-n51-k5-41-42	713.18	166.41	327.52	1495.1	304.14	668.03
16	Instance50-3	1792.46	265.52	881.32	-	-	-
17	Instance50-11	1674.18	291.99	834.06	-	-	-
18	Instance50-20	1691.18	259.75	868.15	3446.14	465.06	1823.6
19	Instance50-26	1612.94	255.79	837.1	3203.12	412.63	1653.27
20	Instance50-31	1798.64	266.49	902.22	3414.9	405.56	1792.65
21	Instance50-35	1790.34	278.92	892.89	-	-	-
22	Instance50-40	1747.12	258.06	892.45	3545.38	453.93	1975.42
23	Instance50-44	1514.32	256.66	794.19	3361.6	453.62	1882.51
24	Instance50-50	1625.18	268.25	835.98	3664.37	473.24	2070.7
25	Instance50-54	1637.1	256.45	846.4	3764.25	461.42	2154.02

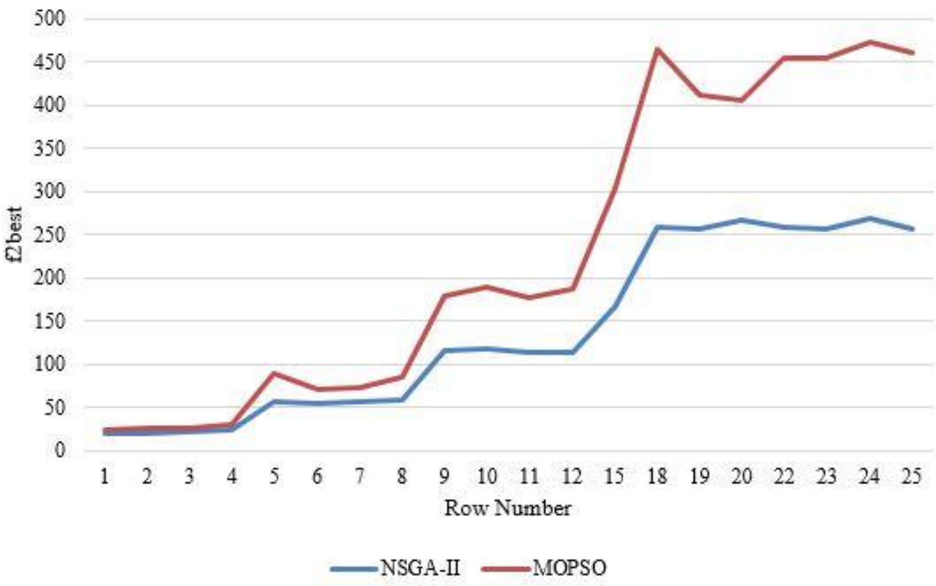

Fig 10. Comparison of f_2^{best} in NSGA-II and MOPSO algorithms

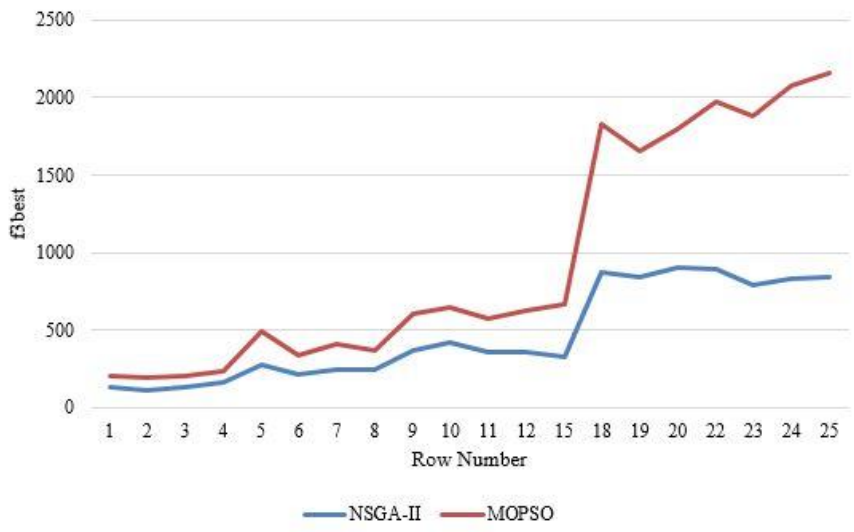

Fig 11. Comparison of f_3^{best} in NSGA-II and MOPSO algorithms

The Pareto fronts for Instance50-50 by NSGA-II and MOPSO algorithm are shown in figures 12 and 13.

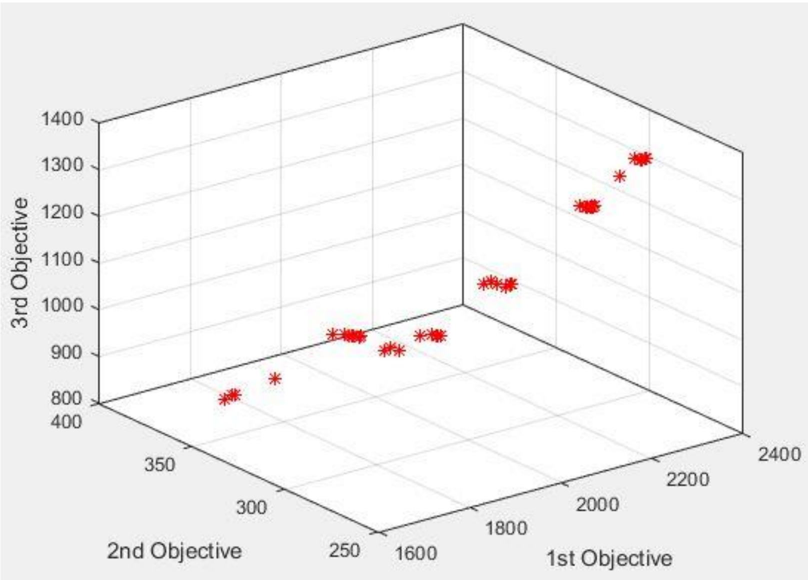

Fig 12. Pareto front for Instance50-50 by NSGA-II

In order to evaluate the performance of the proposed algorithms more than one metric is needed and it cannot be measured adequately with only one performance metric. Hence; for evaluating solutions four NPS, DM, MID and SNS metrics are calculated for Pareto fronts.

The evaluation metrics and Run time for solving the instances are reported in table 6. The values in tables 5 and 6 are means of five repetitions of the algorithms for each instance.

Figure 14 shows the computational time for both algorithms. Obviously the NSGA-II algorithm run time is more than MOPSO. Since, NSGA-II searches more regions of solution space, and finds better number of non-dominated solutions, this higher computational time is reasonable.

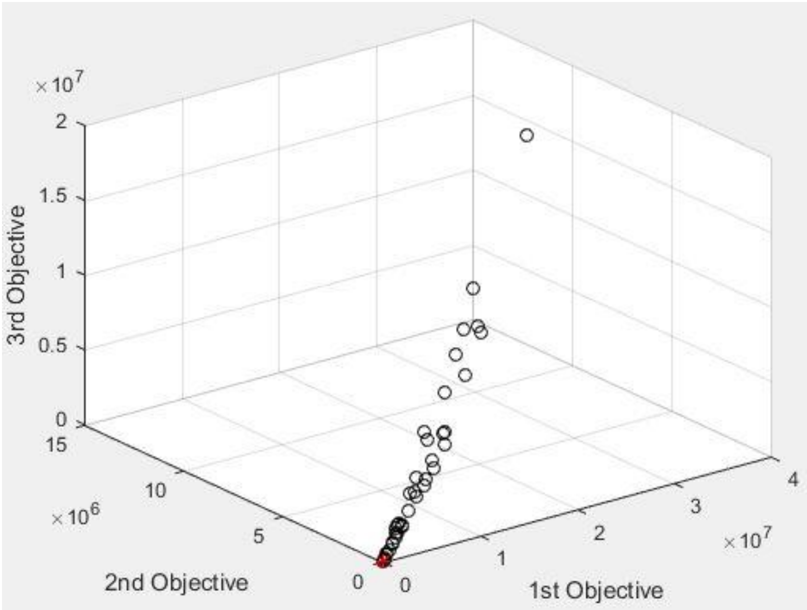

Fig 13. Pareto front for Instance50-50 by MOPSO

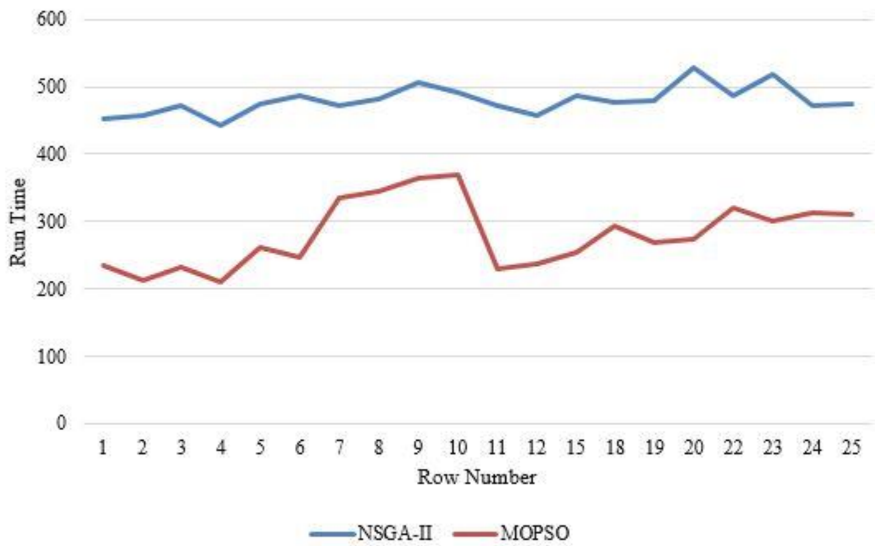

Fig 14. Run times for NSGA-II and MOPSO algorithms

According to table 6 and by comparison of the not empty rows, it can be observed that in all instances NSGA-II has better performance based on DM and MID metrics and also in 95 and 75 percent of the instances based on NPS and SNS metrics, respectively. The empty rows show that the MOPSO algorithm wasn't able to find a feasible solution for the corresponding instances after five algorithm repetition. So, it is concluded that totally, NSGA-II algorithm outperforms MOPSO algorithm.

Table 6. Comparison of two algorithms based on evaluation metrics

row	Instance Name	NSGA-II					MOPSO				
		NPS	DM	MID	SNS	time	NPS	DM	MID	SNS	time
1	E-n13-k4-17	87.8	58.18	312.45	15.64	451.23	69	52.93	468.28	20.21	234.8
2	E-n13-k4-39	89	75.38	318.97	24.50	458.22	16.4	34.91	487.09	45.22	212.15
3	E-n13-k4-48	88.6	89.03	342.20	35.14	472.26	60.4	49.17	499.18	27.54	232.78
4	E-n13-k4-62	89	110.87	434.37	53.74	443.57	34.2	55.41	566.47	49.21	209.89
5	E-n22-k4-s9-19	89	62.75	677.65	20.36	475.32	61.25	28.87	1141.42	5.95	262.67
6	E-n22-k4-s10-14	88.66	35.46	570.27	3.86	485.61	90	32.01	840.09	3.97	247.1
7	E-n22-k4-s13-17	81.33	79.68	645.65	29.81	472.55	82.33	50.34	953.76	16.56	333.84
8	E-n22-k4-s19-21	89	80.05	696.78	30.22	482.83	89	41.79	957.17	8.22	343.59
9	E-n33-k4-s2-13	89	124.38	978.23	72.69	507.39	38	59.23	1465.72	60.01	363.7
10	E-n33-k4-s7-25	88.8	128.21	1043.85	76.35	491.21	72	76.04	1547.74	54.33	368.65
11	E-n33-k4-s16-24	88	65.82	911.01	14.26	472.61	53.5	54.96	1480.37	24.88	230.38
12	E-n33-k4-s22-26	82.4	95.18	940.69	44	458.33	53.25	49.93	1537.45	30.62	237.06
13	E-n51-k5-13-44	86.5	36.06	1044.8	3.77	499.69	-	-	-	-	-
14	E-n51-k5-40-42	86	60.81	1160.1	11.03	519.99	-	-	-	-	-
15	E-n51-k5-41-42	73.4	85.99	868.54	35.85	487.29	29.4	43.86	1751.82	67.87	254.9
16	Instance50-3	86.6	155.65	2151.2	100.01	488.03	-	-	-	-	-
17	Instance50-11	83.6	197.83	2098.28	189.51	517.03	-	-	-	-	-
18	Instance50-20	88.8	205.01	2225.42	187.2	477.53	66.8	66.23	3977.78	39.03	293.79
19	Instance50-26	80.8	198.76	2139.88	189.42	478.51	41.75	63.54	3740.57	58.87	270.15
20	Instance50-31	88.6	194.06	2241.46	156.38	528.56	68	85.24	3998.25	99.11	275.04
21	Instance50-35	88.8	238.63	2342.04	280.82	487.64	-	-	-	-	-
22	Instance50-40	89.2	207.27	2210.76	177.61	487.12	76.8	78.15	4124.18	37.95	321.11
23	Instance50-44	81.2	216.78	2002.04	221.43	519.65	62	77.73	3957.03	38.48	301.84
24	Instance50-50	89.2	218.42	2148.32	190.34	472.72	53.25	66.51	4301.85	49.83	313.6
25	Instance50-54	82.4	199.44	2090.96	171.48	475.55	69.5	46.17	4357.17	12.18	310.48

4-4-Large sized problems

The prodhonce data sets with the link http://prodhonce.free.ir/ are used as large sized instances. These data sets are applicable to the Two Echelon Location Routing Problem (2-ELRP). But by considering vehicles capacity coordinates of the points, the number of customers and satellites and the demand of the customers, we use these data sets for proposed 2E-CVRP problem. These instances cover up to 211 nodes (1 depot and 200 customers and 10 satellites).

These instances and the results of both NSGA-II and MOPSO algorithms are presented in table 7. The MOPSO algorithm is not capable of finding a feasible solution for large sized problems except in two instances. These two instances are in rows 2 and 12 of table 7. While NSGA-II algorithm has the capability of finding a feasible solution in all large sized instances. The values in table 7 are means of five repetitions of the algorithms for each instance. The NSGA-II algorithm has presented better quantities for instances 2 and 12 in comparison with MOPSO algorithm.

5-Conclusion and future research

This paper has presented a tri objective 2E-CVRP problem for perishable products. It aims at minimizing 1) total travel and handling costs, 2) total customers waiting times and 3) total carbon dioxide emissions. The second objective leads to increasing the customer's satisfactions since the products are perishable and the less delivery time cause the more freshness and so more satisfaction. The proposed model is a mixed integer non-linear programming. By applying some linearization methods the problem exchange to the mixed integer linear programming (MILP). The model is solved by Lp-metric method in CPLEX for small sized problems. In order to solving the medium and large sized problems NSGA-II meta-heuristic algorithm is implemented on the model.

Comparing NSGA-II and CPLEX results indicate that the proposed algorithm can find near optimal solutions in much less time. The results of NSGA-II on medium sized problems are compared to the MOPSO results based on four comparison metrics. Based on these metrics the NSGA-II algorithm is capable in finding better Pareto solution but in a more run time versus MOPSO algorithm and by problem size enhancement the difference between two algorithms will increase. In large sized instances MOPSO can find a feasible solution just in two instances, while NSGA-II can find a feasible solution in all instances. By comparing the two instances can find NSGA-II algorithm outperforms MOPSO.

Table 7. Results of NSGA-II on large sized instances

row	Instance Name	NSGA-II			MOPSO		
		f_1^{best}	f_2^{best}	f_3^{best}	f_1^{best}	f_2^{best}	f_3^{best}
1	coord100-5-1	1672.74	274.74	739.56	-	-	-
2	coord100-5-1b	1740.42	247.9	775.5	3234.6	440.08	1521.8
3	coord100-5-2	2118.58	182.08	916.5	-	-	-
4	coord100-5-2b	1346.13	208.18	605.74	-	-	-
5	coord100-5-3	2212.88	188.24	954.58	-	-	-
6	coord100-5-3b	1536.48	236.78	677.33	-	-	-
7	coord100-10-1	2436.78	275.87	1146.92	-	-	-
8	coord100-10-1b	1826.3	329.34	852.14	-	-	-
9	coord100-10-2	2188.95	275.83	1013.49	-	-	-
10	coord100-10-2b	1649.75	308.28	783.99	-	-	-
11	coord100-10-3	1877.77	263.96	893.13	-	-	-
12	coord100-10-3b	1540.03	288.2	715.78	2726.8	541.63	1403.8
13	coord200-10-1	5692.35	650.33	2467.42	-	-	-
14	coord200-10-1b	4095.7	766.91	1764.73	-	-	-
15	coord200-10-2	4734.44	606.69	2080.8	-	-	-
16	coord200-10-2b	3313.37	703.32	1483.62	-	-	-
17	coord200-10-3	4698.3	587.31	2000.08	-	-	-
18	coord200-10-3b	3500.67	686.34	1490.52	-	-	-

In general, the results show the validity and high performance of the NSGA-II algorithm. The results also indicate the efficiency of the proposed algorithm.

The following recommendations can be considered for future study:

- Because of the road traffic in different hours' of a day, different speeds can be considered for the vehicles and solve the problem based on robust optimization programming
- Using the vehicles with different capacities
- Distribute multi products instead of one product
- Solving the model by exact algorithms such as branch and cut
- Extending the two echelon VRP problem to multi echelon VRP
- Generalize the problem to location routing problem

References

Angel-Bello, F., Martínez-Salazar, I., & Alvarez, A. (2013). Minimizing waiting times in a route design problem with multiple use of a single vehicle. *Electronic Notes in Discrete Mathematics*, *41*, 269-276.

Afshar-Nadjafi, B., & Razmi-Farooji, A. (2014). A Comparison of NSGA II and MOSA for Solving Multi-depots Time-dependent Vehicle Routing Problem with Heterogeneous Fleet. *Journal of Optimization in Industrial Engineering*, *7*(16), 65-73.

Baldacci, R., Mingozzi, A., Roberti, R., & Calvo, R. W. (2013). An exact algorithm for the two-echelon capacitated vehicle routing problem. *Operations Research*, *61*(2), 298-314.

Christofides, N., & Eilon, S. (1969). An algorithm for the vehicle-dispatching problem. *Or*, 309-318.

Crainic, T. G., Ricciardi, N., & Storchi, G. (2009). Models for evaluating and planning city logistics systems. *Transportation science*, *43*(4), 432-454.

Crainic, T. G., Perboli, G., Mancini, S., & Tadei, R. (2010). Two-echelon vehicle routing problem: a satellite location analysis. *Procedia-Social and Behavioral Sciences*, *2*(3), 5944-5955.

Chen, H. K., Hsueh, C. F., & Chang, M. S. (2009). Production scheduling and vehicle routing with time windows for perishable food products. *Computers & Operations Research*, 36(7), 2311-2319.

Cuda, R., Guastaroba, G., & Speranza, M. G. (2015). A survey on two-echelon routing problems. *Computers & Operations Research*, *55*, 185-199.

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

Eberhart, R., & Kennedy, J. (1995, October). A new optimizer using particle swarm theory. In *Micro Machine and Human Science, 1995. MHS'95., Proceedings of the Sixth International Symposium on* (pp. 39-43). IEEE.

Mirzapour Al-e-hashem, S. M. J., & Rekik, Y. (2014). Multi-product multi-period Inventory Routing Problem with a transshipment option: A green approach. *International Journal of Production Economics*, *157*, 80-88.

Mahmoodjanloo, M., Esmaili, M., & Hajiaghaei-Keshteli, M. (2016). A Tri-objective model for the school bus routing problem with a new approach in network design. *2nd International Conference on Industrial & Systems Engineering*, (ICISE), 14, 15 Sep. 2016, Mashhad.

Esmaili, M., & Sahraeian, R. (2017). A new Bi-objective model for a Two-echelon Capacitated Vehicle Routing Problem for Perishable Products with the Environmental Factor. *International Journal of Engineering*, (30)4, 523-531.

Fan, J. (2011). The vehicle routing problem with simultaneous pickup and delivery based on customer satisfaction. *Procedia Engineering*, *15*, 5284-5289.

Glover, F., & Woolsey, E. (1974). Converting the 0-1 polynomial programming problem to a 0-1 linear program. *Operations research*, *22*(1), 180-182.

Govindan, K., Devika, K., Jafarian, A., & Khodaverdi, R. (2013). Two-echelon multiple-vehicle location–routing problem with time windows for optimization of sustainable supply chain network of perishable food. *International Journal of Production Economics*, *152*, 9-28.

Holland, J. H. (1975). Adaptation in natural and artificial systems. University of Michigan Press. *Ann Arbor*, *1*(975), 1.

Hemmelmayer, V. C., Cordeau, J. F., & Crainic, T. G. (2012). An adaptive large neighborhood search heuristic for two-echelon vehicle routing problems arising in city logistics. *Computers & operations research*, *39*(12), 3215-3228.

Jepsen, M., Spoorendonk, S., & Ropke, S. (2013). A branch-and-cut algorithm for the symmetric two-echelon capacitated vehicle routing problem. *Transportation Science*, *47*(1), 23-37.

Lin, C., Choy, K. L., Ho, G. T., Chung, S. H., & Lam, H. Y. (2014). Survey of green vehicle routing problem: past and future trends. *Expert Systems with Applications*, *41*(4), 1118-1138.

Maghsoudlou, H., Kahag, M. R., Niaki, S. T. A., & Pourvaziri, H. (2016). Bi-objective optimization of a three-echelon multi-server supply-chain problem in congested systems: Modeling and solution. *Computers & Industrial Engineering*, *99*, 41-62.

Perboli, G., Tadei, R., & Vigo, D. (2011). The two-echelon capacitated vehicle routing problem: models and math-based heuristics. *Transportation Science*, *45*(3), 364-380.

Rabbani, M., Ramezankhani, M. J., Farrokhi-Asl, H., & Farshbaf-Geranmayeh, A. (2015). Vehicle routing with time windows and customer selection for perishable goods. *International Journal of Supply and Operations Management*, *2*(2), 700.

Rabbani, M., Farrokhi-Asl, H., & Asgarian, B. (2017). Solving a bi-objective location routing problem by a NSGA-II combined with clustering approach: application in waste collection problem. *Journal of Industrial Engineering International*, *13*(1), 13-27.

Santos, F. A., Mateus, G. R., & da Cunha, A. S. (2014). A branch-and-cut-and-price algorithm for the two-echelon capacitated vehicle routing problem. *Transportation Science*, *49*(2), 355-368.

Soysal, M., Bloemhof-Ruwaard, J. M., & Bektaş, T. (2015). The time-dependent two-echelon capacitated vehicle routing problem with environmental considerations. *International Journal of Production Economics*, *164*, 366-378.

Song, B. D., & Ko, Y. D. (2016). A vehicle routing problem of both refrigerated-and general-type vehicles for perishable food products delivery. *Journal of Food Engineering*, *169*, 61-71.

Wang, X., Wang, M., Ruan, J., & Zhan, H. (2016). The multi-objective optimization for perishable food distribution route considering temporal-spatial distance. *Procedia Computer Science*, *96*, 1211-1220.

Zeng, Z. Y., Xu, W. S., & Xu, Z. Y. (2013, November). A two-phase hybrid heuristic for the two-echelon vehicle routing problem. In *Chinese Automation Congress (CAC), 2013* (pp. 625-630). IEEE.