

## Solving a robust capacitated arc routing problem using a hybrid simulated annealing algorithm: A waste collection application

E. Babaee Tirkolaee<sup>1\*</sup>, M. Alinaghian<sup>2</sup>, M. Bakhshi Sasi<sup>2</sup>, M. M. Seyyed Esfahani<sup>1</sup>

### Abstract

The urban waste collection is one of the major municipal activities that involves large expenditures and difficult operational problems. Also, waste collection and disposal have high expenses such as investment cost (i.e. vehicles fleet) and high operational cost (i.e. fuel, maintenance). In fact, making slight improvements in this issue lead to a huge saving in municipal consumption. Some incidents such as altering the pattern of waste collection and abrupt occurrence of events can cause uncertainty in the precise amount of waste easily and consequently, data uncertainty arises. In this paper, a novel mathematical model is developed for robust capacitated arc routing problem (CARP). The objective function of the proposed model aims to minimize the traversed distance according to the demand uncertainty of the edges. To solve the problem, a hybrid metaheuristic algorithm is developed based on a simulated annealing algorithm and a heuristic algorithm. Moreover, the results obtained from the proposed algorithm are compared with the results of exact method in order to evaluate the algorithm efficiency. The results have shown that the performance of the proposed hybrid metaheuristic is acceptable.

**Keywords:** Waste collection, CARP, hybrid metaheuristic algorithm, simulated annealing algorithm, robust optimization.

*Received: Mar 2016-09*

*Revised: Jul 2016-04*

*Accepted: Nov 2016-25*

### 1. Introduction

The solid waste production and its corresponding incompatibility (e.g. social, economic and environmental incompatibility) have encountered the management of municipal services with a lot of problems in the field of collection, transportation, processing and disposal of such wastes. Since between 75 and 80 percent of the cost of solid waste material management relates to its collection and transportation (Golden et al. 2001) therefore, evaluation and optimization of this system play a significant role in reducing the municipal services management problems. Waste materials must be collected, transferred and disposed as soon as possible and by complying

\* Corresponding Author.

<sup>1</sup> Department of Industrial Engineering, Mazandaran University of Science and Technology, Babol, Iran.

<sup>2</sup> Department of Industrial Engineering, Isfahan University of technology, Isfahan, Iran.

hygiene recommendations. So that, the best method is to collect them directly from the houses and then, transport them to the disposal facility. Based on above, the importance of waste collection optimal system becomes more evident than before. As a result, choosing the optimal policy for waste collection has a decisive role in cost reduction.

Anyway, there are several kinds of vehicle routing problem to collect the urban waste. Two main categories of routing problem in waste collection are as follows (Beltrami and Bodin 1974): 1) Node Routing Problem: The problem includes commercial waste collection (e.g. the garbage of restaurants, organizations and etc.) such that there are large containers in the main streets of the city which are loaded by the vehicles. In this problem, the containers are considered as nodes, 2) Arc Routing Problem: The problem includes household waste collection in the network of streets and alleys in the city where the wastes are kept in small shelves or garbage bags. Actually, in this problem, the precise location of each customer is not required.

In the household waste collection, the wastes are along the edges. In addition, the capacities of vehicles are limited and they will decrease when they move from one alley to another one or from one arc to another.

Beltrami, E. and Bodin, L. D. (1974) presented one of the first routing problems for the urban waste collection in the form of classical vehicle routing problem (VRP) in order to collect wastes in New York and Washington Municipalities. The problem included the different types of vehicles (i.e. trucks, trolley boats, tug ships, mechanical vacuums). They were able to improve Clarke and Wright algorithm in determining the optimal route. The outcome of their study was implemented in the municipalities and made high advantages for the cities.

Santos et al. (2010) suggested a modified ant colony algorithm to solve the capacitated arc routing problem (CARP). They made some modifications in the initial population, ant decision rule and local search procedure. Finally, they implemented their algorithm for seven standard sets of available examples in CARP literature and obtained appropriate results.

Fleury and Lacomme (2004, 2005) formulated a stochastic arc routing problem (SCARP) for the first time in which some parameters (customers' demands, vehicles travel times, etc.) were stochastic. In the waste collection problem, customers' demands can be random in nature or can be considered as a decision variables.

Laporte et al. (2010) presented a CARP problem considering stochastic demands. In the problem, if the capacity of the vehicles are exceeded, failure will occur in the paths. They solved the problem by a neighborhood search heuristic algorithm.

Chen et al. (2014) presented a CARP to formulate a routing problem for investigating the maintenance activities of a road network. In their problem, traveling and servicing time were stochastic. The model was developed using chance constraint programming and it was solved by a branch and bound algorithm.

Willemse and Joubert (2016) presented a new mathematical model for CARP considering intermediate facilities and time limitation. Furthermore, they developed four heuristics to solve the problem and compared their efficiency. Chen et al. (2016) presented a metaheuristic method (HMA) for CARP. The proposed method was able to solve the problem instances in a reasonable time and improved the solution for 6 classic instances and 9 large size instances. They also investigated the effects of the parameters of the algorithm on the results.

Yu and Lin (2015) developed a metaheuristic for the time-dependent prize-collecting arc routing problem. In their study, objective function is determined by subtracting the route traversing cost from the profit gained from the specific route. Also, the traversing time of the edges depends on the departure time of the vehicles.

Changes in weekly visit patterns for evaluating and choosing the best pattern are the issues that the managers and decision makers of this field are interested in its implementation. But, studying the effects of the pattern before its implementation, requires an optimizer system. On the other

hand, these changes will cause uncertainty in the data. Moreover, the occurrence of unexpected incidents and disasters are the other reasons of uncertainty in the parameters of CARP.

The aim of this paper is to present an innovative mathematical model for the robust CARP in regards with the waste collection problem which aims to minimize the total cost (the usage cost of vehicles and traversing cost of the edges) considering the demand uncertainty of the required edges. Also, the number of required vehicles to service the all required edges can be determined by the proposed model.

With regard to the NP-hardness of the problem (Golden and Wong 1981), a hybrid metaheuristic algorithm is presented based on a simulated annealing and a constructive heuristic algorithm to solve the problem. Additionally, this work adopts the concept of Taguchi method into the proposed algorithm to enhance the solution quality.

The remainder of the current paper is organized as follows; in section 2, the robust optimization approach of Bertsimas and Sim is discussed. In section 3, the mathematical model for robust CARP specific to the urban waste collection is presented. In section 4, the proposed solution method for the problem is presented. Finally, in section 5, the experimental results are presented.

## 2. Robust CARP

The problem includes determining the optimal number of vehicles and the optimal tours for each vehicle with regard to the objective function which minimizes the usage cost of vehicles and the traversing cost of all traversed edges. In this problem, the vehicles are in the depot, firstly. They begin their tours to service the required edges and then, they return to the depot after filling their capacity. The capacity constraint of vehicles results in determining the number of required vehicles. Moreover, demand is uncertain. Thus, the robust model is presented based on the interval robust optimization approach.

### 2.1. Problem assumptions

- The objective function, in addition to the minimization of traversing cost of the edges, is to minimize the number of required vehicles to meet the total demand.
- Each required edge is serviced only by one vehicle. Also, only one depot is considered in the problem.
- The vehicles are homogeneous and the graph network is asymmetric. Each vehicle begins the tour from the depot and returns there finally.
- The traversing time and the cost of routes are the same among the vehicles.
- The demand of required edges fluctuates in the uncertainty interval  $\left[ d_{ij} - \hat{d}_{ij}, d_{ij} + \hat{d}_{ij} \right]$ .

### 2.2. Sets, parameters and variables

In this section, the robust model of the proposed problem is presented. For this purpose, sets, parameters and variables are defined firstly. Then, the theoretical basis of the robust optimization is described and finally, the robust model is presented.

#### Sets

$V$  : Set of the nodes

$K$  : Set of the vehicles

$n$  : Set of the available nodes in the graph network

$E$  : Set of all the edges defined in the network

$E_R$  : Set of all the required edges defined in the network

$S$  : Optional set of all the edges defined in the network

$V[S]$  : Set of the nodes defined in S

### Parameters

$c_{ij}$  : Traversing cost of the edge  $(i, j)$ , such that  $(i, j) \in E$

$d_{ij}$  : Demand of edge  $(i, j)$ , such that  $(i, j) \in E$

$cv_k$  : Cost of  $k$ th vehicle usage

$W$  : Vehicle capacity

$M$  : A very large number

### Decision variables

$x_{ijk}$  : 1 if edge  $(i, j) \in E$  is traversed by vehicle  $k$ , otherwise 0.

$y_{ijk}$  : 1 if edge  $(i, j) \in E$  is serviced by vehicle  $k$ , otherwise 0.

$u_k$  : 1 if vehicle  $k$  is being used such that  $k \in K$ , otherwise 0.

### 2.3. The robustness of capacity constraints of vehicles

In this section, a robust mathematical model for the classic arc routing problem is formulated based on Bertsimas and Sim approach (2003, 2004). In this model, the uncertain parameter  $d_{ij}$  takes value in a symmetric interval  $\left[ d_{ij} - \hat{d}_{ij}, d_{ij} + \hat{d}_{ij} \right]$ , where  $\hat{d}_{ij}$  is the deviation of  $d_{ij}$ . This parameter causes uncertainty in the capacity constraints of vehicles. The corresponding constraints with uncertainty are:

$$\sum_{(i,j) \in E_R} d_{ij} y_{ijk} \leq W, \forall k \in K \quad (1)$$

So, the parameter  $\Gamma_k$  is defined for each vehicle  $k$  such that  $\Gamma_k \in [0, |J_k|]$ , where  $J_k$  is the set of the serviced edges by vehicle  $k$  and it is defined as a set of uncertain coefficients in the constraint corresponding to the capacity constraint of vehicle  $k$ . In fact,  $\Gamma_k$  ensures the robustness adjustment against the conservatism level of solution.

Bertsimas and Sim (2004) showed that it is unlikely that all of the coefficients change simultaneously. Therefore, it is assumed that up to  $\lfloor \Gamma_k \rfloor$  of these coefficients are allowed to change, and one coefficient  $d_{t_k}$  (corresponding to the edge  $t_k$ ) can change up to  $(\Gamma_k - \lfloor \Gamma_k \rfloor) \hat{d}_{t_k}$ . Then, the solution still remains feasible. In other words, it is assumed that only a subset of the coefficients are permitted to affect the solution adversely. This assumption assures that if it happens in a real case, the robust optimal solution will be feasible definitely. Moreover, due to the symmetric distribution of variables, even if more than  $\lfloor \Gamma_k \rfloor$  changes occur, then, the robust solution will be feasible with a very high probability. Thus,  $\lfloor \Gamma_k \rfloor$  is called the protection level for vehicle  $k$ . On this basis, the robust mathematical model of the proposed nominal linear optimization is as follows:

$$\sum_{(i,j) \in E_R} d_{ij} y_{ijk} + \max_{\{s_k \cup \{t_k\} | s_k \subseteq J_k, |s_k| = \lfloor \Gamma_k \rfloor, t_k \in J_k \setminus s_k\}} \left\{ \sum_{(i,j) \in s_k} \sum \hat{d}_{ij} |y_{ijk}| + (\Gamma_k - \lfloor \Gamma_k \rfloor) \hat{d}_{t_k} |y_{t_k k}| \right\} \leq W \quad \forall k \in K \quad (2)$$

Where:

$$\beta_k(y_{ijk}, \Gamma_k) = \max_{\{s_k \cup \{t_k\} | s_k \subseteq J_k, |s_k| = \lfloor \Gamma_k \rfloor, t_k \in J_k \setminus s_k\}} \left\{ \sum_{(i,j) \in s_k} \sum \hat{d}_{ij} |y_{ijk}| + (\Gamma_k - \lfloor \Gamma_k \rfloor) \hat{d}_{t_k} |y_{t_k k}| \right\} \quad (3)$$

Equation (3) is equal to the optimal value of the problem below which is linear (For more information see (2004)).

$$\begin{aligned} \beta_k(y_{ijk}, \Gamma_k) &= \max \sum_{(i,j) \in J_k} \sum \hat{d}_{ij} |y_{ijk}| z_{ijk} \\ \sum_{(i,j) \in J_k} z_{ijk} &\leq \Gamma_k \quad , \forall k \in K \\ 0 \leq z_{ijk} &\leq 1 \quad , (i,j) \in J_k , \forall k \in K \end{aligned} \quad (4)$$

$\Gamma_k$  Can be integer or not. The point is, if  $\Gamma_k = 0$ , then  $\beta_k(y_{ijk}, \Gamma_k) = 0$ . Thus, constraint  $k$  would be similar to the  $k$ th constraint of the nominal problem. Likewise, if  $\Gamma_k = |J_k|$ , Bertsimas and Sim approach will be as well as Soyster's method. Therefore, by changing  $\Gamma_k$  in  $[0, |J_k|]$ , the robustness adjustment is ensured against the conservatism level of the solution. Anyway, the dual of the problem is as follows:

$$\begin{aligned} \text{Dual } \beta_k(y_{ij}, \Gamma_k) &= \min \sum_{(i,j) \in J_k} r_{ij} + \Gamma_k z_k \\ \text{Subject To:} \\ z_k + r_{ijk} &\geq \hat{d}_{ij} |y_{ijk}| \quad \forall k , \forall (i,j) \in J_k \\ r_{ijk} &\geq 0 \quad \forall k , \forall (i,j) \in J_k \end{aligned} \quad (5)$$

Finally, the linear and robust form of the problem would be:

$$\begin{aligned} \sum_{(i,j) \in J_k} d_{ij} y_{ijk} + z_k \Gamma_k + \sum_{(i,j) \in J_k} p_{ijk} &\leq W_k \quad \forall k \in K \\ z_k + r_{ijk} &\geq \hat{d}_{ij} E_{ijk} , -E_{ijk} \leq y_{ijk} \leq E_{ijk} \quad \forall k \in K , \forall (i,j) \in J_k \\ r_{ijk} &\geq 0 , z_k \geq 0 \quad \forall k \in K , \forall (i,j) \in J_k \end{aligned} \quad (6)$$

As it is assumed that all of the required edge are uncertain, so, the set  $J_k$  includes a subset of the required edges ( $E_R$ ) which are being serviced by vehicle  $k$ .

## 2.4. Proposed robust model

In this section, the proposed robust mathematical model based on the developed model in (Moshe and Langevin 2000) is presented:

$$\text{MinimizeCost}_{total} = \sum_{(i,j) \in E} \sum_{k \in K} c_{ij} x_{ijk} + \sum_{k \in K} cv_k u_k \quad (7)$$

*Subject To :*

$$\sum_{(i,j) \in E} x_{ijk} = \sum_{(j,i) \in E} x_{jik}, \forall i \in V, \forall k \in K \quad (8)$$

$$\sum_{k \in K} (y_{ijk} + y_{jik}) = 1, \forall (i,j) or (j,i) \in E_R \quad (9)$$

$$\sum_{(i,j) \in E_R} d_{ij} y_{ijk} + z_k \Gamma_k + \sum_{(i,j) \in E_R} r_{ijk} \leq W, \forall k \in K \quad (10)$$

$$z_k + r_{ijk} \geq \hat{d}_{ij} E_{ijk}, \forall k \in K, \forall (i,j) \in E_R \quad (11)$$

$$-E_{ijk} \leq y_{ijk} \leq E_{ijk}, \forall k \in K, \forall (i,j) \in E_R \quad (12)$$

$$y_{ijk} \leq x_{ijk}, \forall (i,j) \in E, \forall k \in K \quad (13)$$

$$\sum_{(i,j) \in E} x_{ijk} \leq M u_k, \forall k \in K \quad (14)$$

$$\sum_{i,j \in S} x_{ijk} \leq 1 + n h_k^S, \forall S \subseteq V \setminus \{1\}; S \neq \emptyset; \forall k \in K \quad (15)$$

$$\sum_{i \in S} \sum_{j \neq S} x_{ijk} \geq 1 - f_k^S, \forall S \subseteq V \setminus \{1\}; S \neq \emptyset; \forall k \in K \quad (16)$$

$$h_k^S + f_k^S \leq 1, \forall S \subseteq V \setminus \{1\}; S \neq \emptyset; \forall k \in K \quad (17)$$

$$h_k^S \in \{0,1\}, f_k^S \in \{0,1\}, \forall S \subseteq V \setminus \{1\}; S \neq \emptyset; \forall k \in K \quad (18)$$

$$x_{ijk} \in \{0,1\}, y_{ijk} \in \{0,1\}, u_k \in \{0,1\}, \forall (i,j) \in E, \forall k \in K \quad (19)$$

$$E_{ijk}, z_k, r_{ijk} \geq 0 \quad (20)$$

The objective function includes minimizing the total traversing and usage cost of the vehicles. Constraints (8) are the flow balance equations for each vehicle. These constraints control the input and output flows of the edges (i.e. if a vehicle enters a node, it must leave it necessarily). Constraints (9) ensure that any required edges must be serviced. Constraints (10) to (12) are the corresponding constraints to the robustness. Constraints (13) express that a required edges will be serviced by the vehicle which has traversed the edge (in other words, the vehicle may traverse the edge without servicing). Constraints (14) ensure that  $k$ th vehicle will be used, whenever its usage cost is paid. Constraints (15) to (17) are the sub-tour elimination constraints. Constraints set (18) to (20) specify the types of variables.

## 2.5. Determining a probability bounds for constraint violation

In the proposed robust model, if the probability of violating the capacity constraint for each vehicle be  $\alpha$ , then the corresponding constraint to the capacity for each vehicle must satisfy the following equation:

$$P\left(\sum_{(i,j) \in E_R} d_{ij} y_{ijk} \leq W_k\right) \geq (1 - \alpha) \quad (20)$$

In continuation, a proposition is presented which is used for determining a bound for the above probability (Yu, and Lin 2015).

**Preposition 1.** Let  $x^*$  be an optimal solution for the robust problem. The probability that the  $i$ th constraint is violated holds true in:

$$pr\left(\sum_j \tilde{a}_{ij} x_j^* > b_i\right) \leq B(n, \Gamma_i) \quad (21)$$

Where:

$$B(n, \Gamma_i) = \frac{1}{2^n} \left\{ (1 - \mu) \sum_{l=v}^n \binom{n}{l} + \mu \sum_{l=v+1}^n \binom{n}{l} \right\} = \frac{1}{2^n} \left\{ (1 - \mu) \binom{n}{v} + \sum_{l=v+1}^n \binom{n}{l} \right\} \quad (22)$$

Where  $n = |J_i|$ ,  $v = \frac{\Gamma_i + n}{2}$  and  $\mu = v - \lfloor v \rfloor$ . Therefore, for the proposed problem, firstly the number of serviced edges in each route is specified ( $n$ ) to control the violation probability of the capacity constraints. Then,  $\Gamma_k$  is determined based on the desired service level and the proven equation. Thus,  $\Gamma_k$  should be greater than the least value which satisfies equation (22). In this problem, service level is 0.95.

## 3. Solution method

Due to the high degree of the problem complexity and also, considering uncertainty in the constraints of the proposed model, solving the problem by the exact method is only possible in small size problems. Therefore, in this section, a heuristic algorithm is developed to generate some initial solutions and then, a simulated annealing algorithm is applied to provide proper feasible solutions. Afterward, to evaluate the efficiency of the proposed algorithms, the CPLEX solver is used. In next sections, the mechanism of the proposed algorithm will be discussed.

### 3.1. Solution representation

In metaheuristic algorithms, the solution representation method has profound effects on the quality of results. In this paper, a string is used to represent the traversing sequence of required edges. In other words, the applied chromosome here is a sequence of the required edges. Also, the performance of the string is described. First, the vehicle begins its tour from the depot and returns there after traversing the specified required edges in the string (after filling out the capacity). Then, the next vehicle starts from the depot and services the next remained required edges in the string. This will continue until all the required edges are being serviced. Also, the traversing cost between two required edges is the shortest path between them which has been determined by dijkstra algorithm before.

### 3.2. The heuristic algorithm

A constructive heuristic algorithm is used to generate a number of initial solutions. The steps of algorithm are as follows:

- 1- Start from the first vehicle.
- 2- Select  $k$  required edges which have the least distance to the depot. Select one of them randomly and then service it.
- 3- After servicing the selected edge, select  $k$  edges with the least distance from the selected node. Select one of them randomly and service it.
- 4- If all the required edges are being serviced, go to step 5. Otherwise go to step 3.
- 5- Stop the algorithm. Report the solution.

Furthermore, with regard to the solution of some random instances, the number of generated solutions by heuristic method is considered equal to the number of required edges of the network.

### 3.3. Simulated annealing algorithm

In this research, simulated annealing algorithm is applied to improve the quality of the solutions (Kirkpatrick and Vecchi 1983). As a consequence, all the initial solutions are improved individually by this algorithm. General framework of SA is such that it starts with an initial solution. The initial parameters of the algorithm include the number of iterations in each temperature ( $M$ ), initial temperature ( $T_0$ ), cooling rate ( $\alpha$ ), final temperature ( $T_{end}$ ) and Boltzman constant ( $K$ ) which are initialized before the search starts. Then, a neighborhood is created for the initial solution and it will be replaced with the initial solution, if its objective function value is better than the initial solution. Otherwise, the difference between two objective function values is calculated ( $\Delta$ ) and a random number is generated from the interval [0, 1]. In continuation, the random generated number is compared with the probability equation of the algorithm ( $\exp(-\Delta / KT)$ ) and the worse solution is accepted, if the random number is less than the value of probability equation. In each temperature, a number of iterations is run and then, the temperature is reduced. The temperature reduction and cooling equation is  $\alpha T \rightarrow T$ . In addition, stopping criterion is to reach to the final temperature.

The local searches have been used in this algorithm are as below:

1. Two tours are selected randomly. If there is a mutual edge/edges between tours, one of them is selected randomly and so, both tours will be divided into the two parts accordingly. The first part of the first tour is combined with the second part of the second tour and the two other components of the tours are combined with each other. As a result, two new tours are composed.
2. Two tours are selected randomly. If there are two mutual edges between them, the sequence between the two mutual edges are changed together.
3. An edge of a tour is selected randomly and its direction is reversed.
4. A part of a tour is selected randomly and their directions are changed.

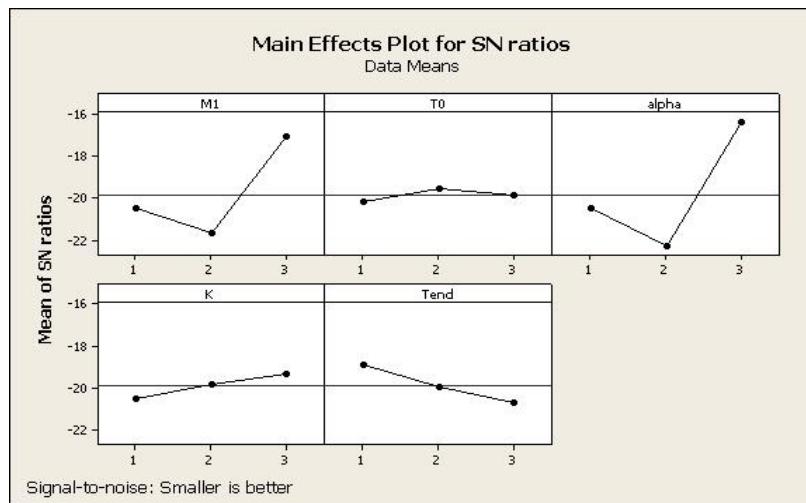
### 3.4. Design of experiments: Taguchi method

In this research, Taguchi method (Taguchi et al. 2005) is utilized to adjust the parameters of the proposed algorithm. Taguchi improved a family of fractional factorial testing matrices such that he was able to implement a design of experiment method, after numerous experiments, in which the number of experiments is reduced for a problem. Since the less values of objective function and run time is more appropriate therefore, signal to noise ratio (S/N Ratio) is calculated as:

$$S/N \text{ Ratio} = -10 \log \sum_{i=1}^k (y_i^2 / k) \quad (23)$$

Where  $y_i$  and  $k$  indicate the performance of experiment  $i$  and the total iterations of experiment, respectively. Maximizing the S/N ratio leads to the minimization of dispersion of the objective function. In the proposed algorithm, there are five variables which are specified as controllable factors: number of iterations in each temperature ( $M$ ), initial temperature ( $T_0$ ),

cooling rate ( $\alpha$ ), final temperature ( $T_{end}$ ) and Boltzman constant ( $K$ ). Also, to adjust the parameters, three levels are considered for each of them and  $L27$  is selected according to the number of the parameters and the levels of design of experiments. Three levels have been determined for each factor (parameter) to design the experiments. In Table 1, the first column represents all of the parameters of SA and the next three columns represent the considered initial values for these parameters. These values have been chosen according to the proposed values in (Glover and Kochenberger 2005). After implementing experiments on the data of Table 1, the average of S/N ratio for 27 levels of Taguchi experiment has been shown in Figure 1. Also, the optimal values of parameters for SA algorithm have been presented in the last column of Table 1.



**Figure 1: Average of S/N ratio for parameter adjustment of SA algorithm**

**Table 1: Parameter values of SA**

Parameter	Initial values			Optimal values
	1	2	3	
$T_0$	150	200	300	200
$T_{end}$	1	2	5	1
$\alpha$	0.8	0.95	0.98	0.98
$M$	2	5	10	10
$K$	0.2	0.5	0.8	0.8

### 3.5. Calculating fitness function of generated solutions

To determine the fitness function value of each generated solution by metaheuristic method, firstly the objective function value (which is the total cost of traversed distance by vehicle) is calculated regardless to the solution robustness. Then, the following steps are done:

**Step 1.** The number of required edges in each constructed tour are considered. Then, on the basis of equation (20), (21) and the defined service level,  $\Gamma_k$  is determined for the vehicles.

**Step 2.**  $\Gamma_k$  number of edges with the highest uncertainty value of demand are selected and their demands are increased into the maximum possible value, according to their demand uncertainty interval.

**Step 3.** If  $\Gamma_k$  is non-integer, the demand of the last selected edge is increased by  $(\Gamma_k - \lfloor \Gamma_k \rfloor) \hat{d}$ .  $\hat{d}$  is defined by multiplying the demand parameter by the deviation level  $\delta$  (i.e.,  $\hat{d} = \delta d$  ).

**Step 4.** With regard to the considered changes in the demand of the edges, if the demand of each edge exceeds the vehicle capacity, a penalty proportional to the violation value is added to the objective function.

## 4. Numerical results

In this section, the method of generating different random instances for evaluating the performance of the proposed algorithm is described. Also, the efficiency of algorithm is studied in comparison with CPLEX solver as exact method.

### 4.1. Generation of random instances

To generate the random instances in different size,  $n$  nodes are generated randomly in a two-dimensional coordinate space and node 1 is considered as a depot. Then, a  $n \times n$  matrix is considered to define the available edges in the instances. The value of each cell is one by the probability of 0.7 or zero by the probability of 0.3. One in the matrix, indicates that the edge exists in the instance. Next, 0.7 of the edges are selected randomly as required edges and their demands are generated randomly using uniform distribution  $U[1,4]$ . The capacities and number of vehicles are coi data are given in Table 2.

**Table 2: Random generated instances**

#P	TE	RE	AV	#P	TE	RE	AV
P1	8	6	2	P6	50	40	7
P2	15	11	3	P7	60	50	8
P3	20	15	4	P8	70	58	9
P4	30	22	5	P9	80	65	9
P5	40	30	6	P10	90	72	10

In Table 2, TE, RE and AV are the number of total edges, required edges and available vehicles, respectively.

### 4.2. Efficiency of the proposed algorithm

In this section, the efficiency of the proposed algorithm is studied. The results obtained by solving the robust model by the proposed algorithm and CPLEX solver, are presented in Tables 3 and 4 for two different deviation level of 0.05 and 0.1. CPLEX contains a primal simplex algorithm, a dual simplex algorithm, a network optimizer, an interior point barrier algorithm, a mixed integer algorithm and a quadratic capability. CPLEX, in turn, contains an infeasibility finder. For the problems with the integer variables, CPLEX uses a branch and bound algorithm. A 3600 seconds run time limitation is considered as the stopping criteria and the best obtained solution during this time is reported.

The GAP between the results of the proposed algorithm and CPLEX is calculated by equation (24).

$$GAP = (\text{SA objective} - \text{CPLEX objective}) / \text{CPLEX objective} \quad (24)$$

**Table 3: Comparing results of SA and CPLEX for robust problem with  $\hat{d} = 0.05d$** 

#P	CPLEX		SA		
	Objective function	Run time (sec)	Objective function	Run time (sec)	GAP (%)
<b>P1</b>	998	40.4	1016	8.5	1.8
<b>P2</b>	1121	92.0	1154	15.32	2.92
<b>P3</b>	1815	183.5	1865	18.95	2.73
<b>P4</b>	2689	224.1	2743	26.22	2.01
<b>P5</b>	3550	382.6	3644	34.64	2.65
<b>P6</b>	4144	686.2	4211	42.96	1.61
<b>P7</b>	5318	956.6	5428	51.13	2.06
<b>P8</b>	6180	1825.3	6286	60.3	1.72
<b>P9</b>	6756	3014.5	6845	68.17	1.31
<b>P10</b>	7423	3600	7583	77.42	2.16
<b>Ave</b>	3999.4	1100.52	4077	40.36	2.10

**Table 4: Comparing results of SA and CPLEX for robust problem with  $\hat{d} = 0.1d$** 

#P	CPLEX		SA		
	Objective function	Run time (sec)	Objective function	Run time (sec)	GAP (%)
<b>P1</b>	1021	48.7	1039	10.03	1.75
<b>P2</b>	1609	123.2	1639	15.9	1.85
<b>P3</b>	2572	214.8	2637	22.54	2.53
<b>P4</b>	3160	326.1	3236	30.29	2.40
<b>P5</b>	3695	425.5	3773	38.15	2.11
<b>P6</b>	4594	720.3	4725	45.92	2.85
<b>P7</b>	5450	1106.5	5499	53.4	0.89
<b>P8</b>	6695	1945.0	6809	63.82	1.71
<b>P9</b>	7278	3320.7	7449	72.3	2.35
<b>P10</b>	8315	3600	8461	82.69	1.75
<b>Ave</b>	4438.9	1183.08	4526.6	43.504	2.02

Results have shown that the proposed algorithm has appropriate efficiency in a reasonable solution time. Average gaps for the random instances with  $\hat{d} = 0.05d$  and  $\hat{d} = 0.1d$  are 2.1% and 2.02%, respectively. Also, the maximum run time for this algorithm is 82 seconds (P10). The run times of the proposed algorithm and the exact method are shown in Figure 2.

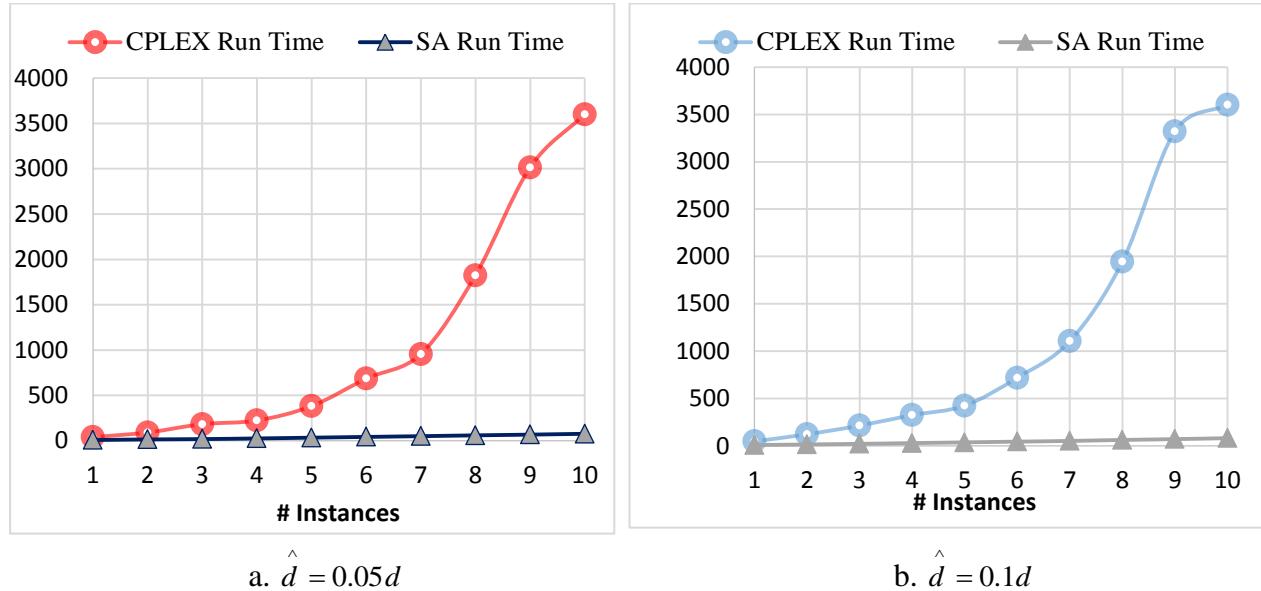


Figure 2: Comparing run time of SA with CPLEX for solving robust model

As it is clear in Figure 2, deviation level does not affect the run time of the proposed algorithm and the exact method significantly. Also, by increasing the size of the problem, run time of the exact method increases exponentially. But, the run time of the proposed algorithm increases by a linear trend.

As an example, the solutions obtained by CPLEX and SA for P1 and  $\hat{d} = 0.05d$  are compared through the figures below. Figure 3 shows a schematic description of P1 including 8 edges with 6 required edges. The solutions obtained by CPLEX and SA have two different routes (Figure 4 and Figure 5), which is the reason why CPLEX reports a little better objective value.

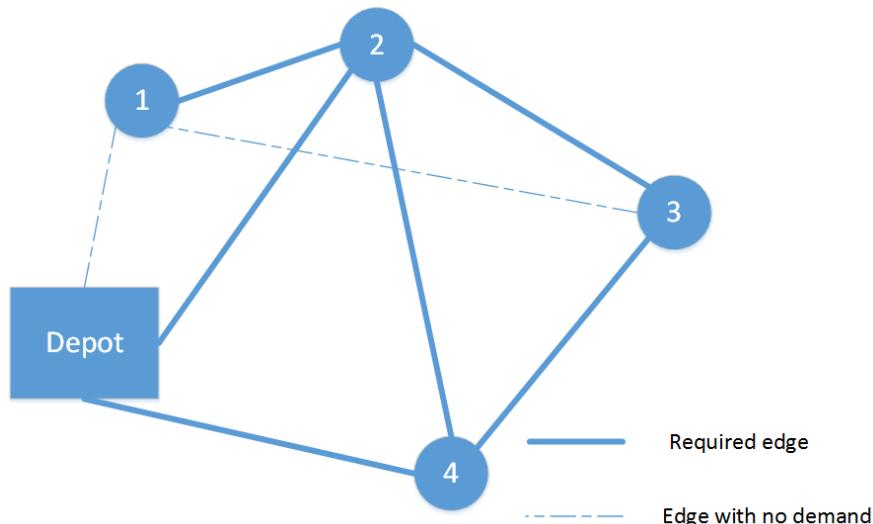


Figure 3: Schematic scheme of P1

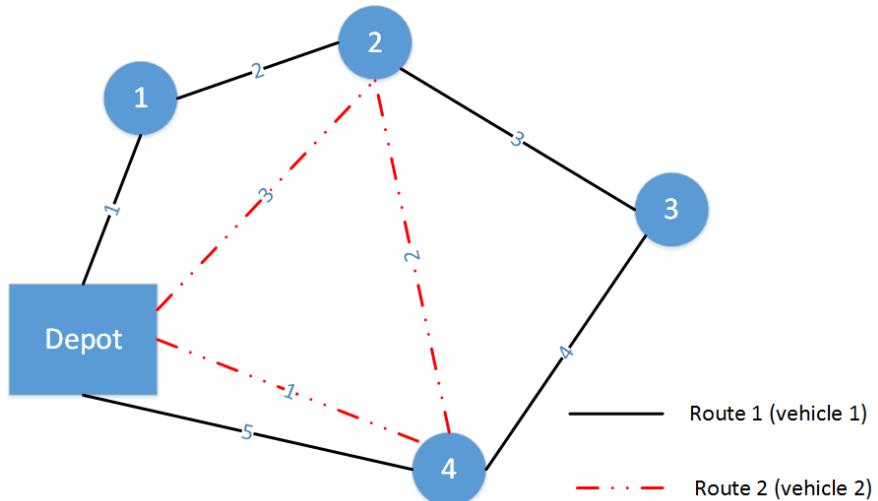


Figure 4: CPLEX solution for P1

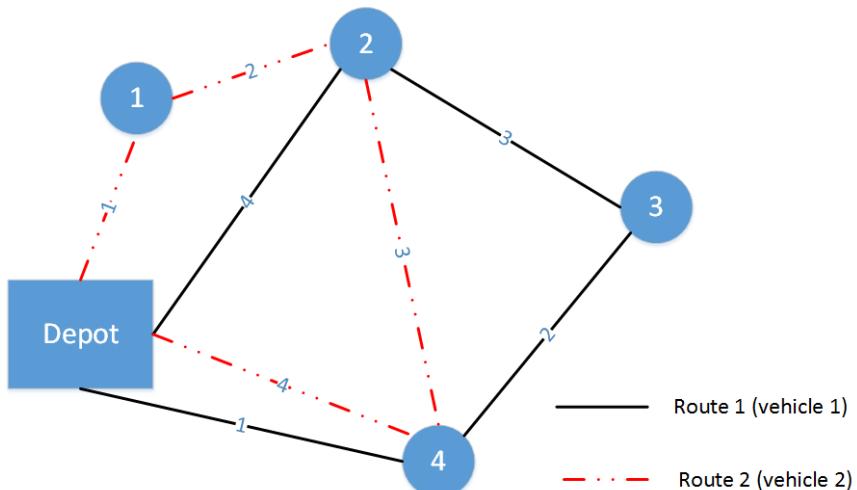
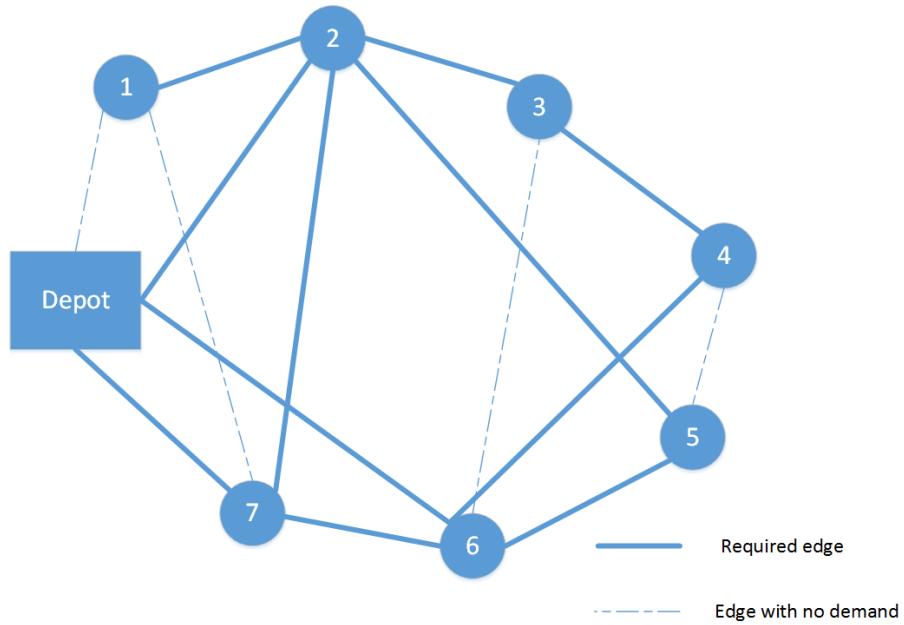


Figure 5: SA solution for P1

The solutions of CPLEX for P2 (as an example) are shown in Table 5 in order to demonstrate the differences between two different deviation levels. A schematic description of P2 is presented in Figure 6.

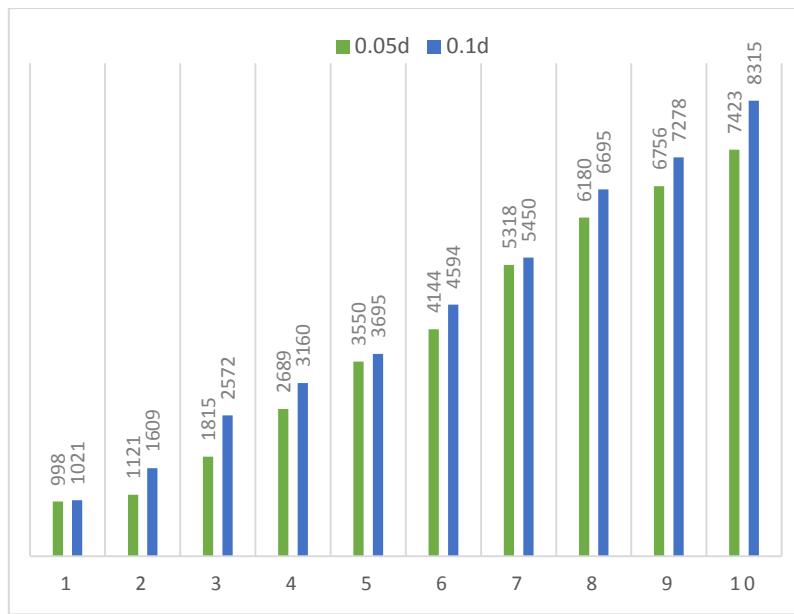


**Figure 6: Schematic scheme of P2**

**Table 5: Solutions of P2 for two different deviation level**

$\hat{d} = 0.05d$	Depot- 1- 2- 5- 6- 7- 2- 3-6- Depot :Route1 (vehicle 1) Route 2 (vehicle 2): Depot- 7- 6- 5- 4- 3- 2- Depot
$\hat{d} = 0.1d$	Depot- 2- 3- 4- 6- 7- Depot:Route 1 (vehicle 1) Route 2 (vehicle 2): Depot- 1- 2- 5- 2- 7- Depot Depot- 6- Depot:Route 3 (vehicle 3)

According to Table 5, the obtained solution of  $\hat{d} = 0.1d$  contains three routes with three active vehicles, while the obtained solution of  $\hat{d} = 0.05d$  contains two routes with two active vehicles and lower total cost. As a result, objective value highly depends on the demand parameter. Figure 7 shows the objective function value of random instances in deviation level of 0.05 and 0.1.

**Figure 7: Objective function values with different deviation level**

By increasing the deviation level from 0.05 to 0.1, the objective function value of random instances increases by 0.15, in average. In other words, if the manager is interested in protection level of 0.1 than 0.05, he should pay 0.15 more cost, in average. Moreover, the maximum increase of objective function value occurs in problems 2 and 3. In these problems, by increasing the deviation level by 0.05, the objective function value increases about 0.4%.

## 5. Conclusion

The optimal routing and allocation of vehicles in CARP is one of the major decisions for organizations such as municipalities in the context of the urban waste collection. The reason is, by optimal allocation and routing of the vehicles they can save vast sum of money. Changes in weekly visit patterns, the occurrence of unexpected incidents and disasters are some reasons which can cause uncertainty in the parameters of CARP. In this paper, a robust mix integer programming model for CARP is developed by taking the demand uncertainty into account. To solve large size problems, a hybrid metaheuristic is used based on the simulated annealing algorithm and a constructive heuristic method. The results showed that the proposed metaheuristic has appropriate efficiency. On the other hand, it was concluded that the problem is sensitive to the deviation level. Therefore, by increasing the deviation level of demand by 0.05, the objective function value increased about 0.15 averagely and it increased even to 0.4% in some problems. For the future studies, more assumptions (like multi trip vehicles, heterogeneous vehicles, etc.) can be added to the problem to make it more applicable in the real world.

## References

- Beltrami, Edward J., and Lawrence D. Bodin., 1974. "Networks and vehicle routing for municipal waste collection." Networks 4.1: 65-94.
- Bertsimas, Dimitris, and Melvyn Sim., 2003."Robust discrete optimization and network flows." Mathematical programming 98.1-3: 49-71.
- Bertsimas, Dimitris, and Melvyn Sim., 2004. "The price of robustness." Operations research 52.1: 35-53.
- Chen, Lu, et al., 2014. Optimizing road network daily maintenance operations with stochastic service and travel times." Transportation Research Part E: Logistics and Transportation Review 64: 88-102.

- Chen, Yuning, Jin-Kao Hao, and Fred Glover., 2016."A hybrid metaheuristic approach for the capacitated arc routing problem." European Journal of Operational Research.
- Dror, Moshe, and André Langevin, 2000.Transformations and exact node routing solutions by column generation. Springer US,
- Fleury, Gérard, Philippe Lacomme, and Christian Prins, 2004. "Evolutionary algorithms for stochastic arc routing problems." Applications of Evolutionary Computing. Springer Berlin Heidelberg. 501-512.
- Fleury, Gérard, et al., 2005."Stochastic capacitated arc routing problems." Document de travail interne. Article en cours de rédaction
- Golden, B., Wong, R., 1981." Capacitated arc routing problems." Network, vol.11, pp.305–315.
- Golden, Bruce L., Arjang A. Assad, and Edward A. Wasil., 2001. "Routing vehicles in the real world: applications in the solid waste, beverage, food, dairy, and newspaper industries." The vehicle routing problem. Society for Industrial and Applied Mathematics.
- Glover, F.W., Kochenberger, G., 2005. "Handbook of metaheuristics", Kluwer Academic Publishers, Norwell,
- Kirkpatrick, S., & Vecchi, M. P., 1983. Optimization by simmulated annealing. science, 220(4598), 671-680.
- Laporte, Gilbert, Roberto Musmanno, and Francesca Vocaturo., 2010. "An adaptive large neighbourhood search heuristic for the capacitated arc-routing problem with stochastic demands." Transportation Science 44.1: 125-135.
- Santos, Luís, João Coutinho-Rodrigues, and John R. Current., 2010. "An improved ant colony optimization based algorithm for the capacitated arc routing problem." Transportation Research Part B: Methodological 44.2: 246-266.
- Taguchi, Genichi, Subir Chowdhury, and Yuin Wu., 2005. Taguchi's quality engineering handbook. Wiley-Interscience,.
- Vincent, F. Yu, and Shih-Wei Lin., 2015. Iterated greedy heuristic for the time-dependent prize-collecting arc routing problem." Computers & Industrial Engineering 90: 54-66.
- Willemse, Elias J., and Johan W. Joubert, 2016. Constructive heuristics for the Mixed Capacity Arc Routing Problem under Time Restrictions with Intermediate Facilities." Computers & Operations Research 68: 30-62.