

**ENGİN PEKEL**, Ph.D. Candidate,<sup>1</sup>  
 E-mail: enginpekel@hitit.edu.tr, pekelc@hotmail.com  
**SELİN SONER KARA**, Ph.D.<sup>2</sup>  
 (Corresponding author)  
 E-mail: ssoner@yildiz.edu.tr

<sup>1</sup> Faculty of Engineering  
 Industrial Engineering Department  
 Hitit University, 19030, Çorum, Turkey

<sup>2</sup> Faculty of Mechanical Engineering  
 Industrial Engineering Department  
 Yıldız Technical University  
 A-Block, 34300, Besiktas, İstanbul, Turkey

Intelligent Transport Systems (ITS)  
 Preliminary Communication  
 Submitted: 6 July 2017  
 Accepted: 10 July 2018

# SOLVING CAPACITATED LOCATION ROUTING PROBLEM BY VARIABLE NEIGHBORHOOD DESCENT AND GA–ARTIFICIAL NEURAL NETWORK HYBRID METHOD

## ABSTRACT

*This paper aims to find the optimal depot locations and vehicle routings for spare parts of an automotive company considering future demands. The capacitated location-routing problem (CLRP), which has been practiced by various methods, is performed to find the optimal depot locations and routings by additionally using the artificial neural network (ANN). A novel multi-stage approach, which is performed to lower transportation cost, is carried out in CLRP. Initially, important factors for customer demand are tested with an univariate analysis and used as inputs in the prediction step. Then, genetic algorithm (GA) and ANN are hybridized and applied to provide future demands. The location of depots and the routings of the vehicles are determined by using the variable neighborhood descent (VND) algorithm. Five neighborhood structures, which are either routing or location type, are implemented in both shaking and local search steps. GA-ANN and VND are applied in the related steps successfully. Thanks to the performed VND algorithm, the company lowers its transportation cost by 2.35% for the current year, and has the opportunity to determine optimal depot locations and vehicle routings by evaluating the best and the worst cases of demand quantity for ten years ahead.*

## KEY WORDS

*artificial neural network; capacitated location-routing problem; genetic algorithm; heuristics; k-nearest neighborhood; variable neighborhood descent;*

## 1. INTRODUCTION

Nowadays, it is obvious that companies should make strategic and operational decisions to optimize and operate their systems more efficiently. In supply chain and logistics management, generating the distribution network is one of the most prominent problems because it presents a great opportunity to bring the cost down and to enhance service quality. The capacitated location-routing problem (CLRP) deals

with the placement of facility locations and routing between the customer and facilities or depots simultaneously by considering that the capacities of depots and vehicles are not violated [1]. The important steps in forming the distribution network are the placement of the locations, such as warehouses, depots, and distribution centers, and routing in a way that considers some given depot or vehicle capacity constraints in order to satisfy customer demands and minimize routing costs, vehicle fixed costs, and depot fixed and operating costs [2].

The determination of demand and capacity issues are important points. In general, most papers choose the demand as deterministic [1-4]. However, that does not offer quite realistic results. In recent years, stochastic and fuzzy demand assumptions have been given much attention in efforts to reach accurate and realistic results. The capacity constraint has been applied in most studies by considering vehicle or depot capacities [1-4]. In reality, the capacity constraint in a vehicle or depot has a great importance. The capacity constraint has been taken into higher consideration because papers dealing with the location-routing problem (LRP) are generally based on real cases [3, 4].

In light of the above, we aim to predict the demand to find out the best possible depots and routes by considering the genetic algorithm artificial neural networks (GA-ANN) approach considering capacity constraint of depot(s) and vehicles for ten years ahead. The reason is that the condition of any constraint, such as the customer demand and the transportation cost, can change in time.

The main contributions of the paper are as follows:

- The consideration of GA-ANN in CLRP offers a different concept by predicting the customer demand. Performing GA-ANN in CLRP provides reliable information about the future demands of customers

- A novel multi-stage approach, which is performed to lower transportation cost, is carried out in CLRP
- This paper offers a method to choose optimal depots and routings by evaluating the best, the most probable, and the worst cases of customer demand for the following years
- The company benefits from the determination of depot locations and routings through cost reduction
- The factors that may affect the customer demand are presented by using a univariate analysis in the paper
- The first neighborhood structure is slightly different, and the others are modified from the literature. The first neighborhood structure is applied by considering the second neighborhood structure simultaneously.

The rest of the paper is organized as follows: Section 2 provides a detailed literature review. Section 3 presents the information about the case study. In addition, the inputs and the output of the GA-ANN model and the classification of inputs with k-NN are presented in this section. Section 4 presents the conceptual framework of the whole research. Section 5 presents the GA-ANN methodology that is applied to predict the demands for the next ten years, and it includes the best prediction parameters and tuning of GA-ANN with regard to  $R^2$ . Section 6 provides the numerical results of GA-ANN calibration and classification. Section 7 presents the LRP model that is dealing with the case study. Section 8 presents the Variable Neighborhood Descent (VND) methodology that is applied to determine the location of depots and routings of the customers. Section 9 presents the result of the routings and locations with regard to VND. Finally, all findings are summarized in the last section.

## 2. LITERATURE REVIEW

The first steps of dealing with LRP date back to the 1960s. However, the models conceived in the 1960s were not similar to the current LRP, since the trip from the last customer on the route back to the starting facility was not regarded seriously. The first real LRP models were developed in the late 1970s and early 1980s through the efforts of various researchers [13].

LRP has several types, such as CLRP, disruption problem, many-to-many LRP, location routing inventory, round trip location, deterministic LRP, and stochastic LRP [17]. Most papers choose the assumption of deterministic demand, which does not offer quite realistic results. In recent years, stochastic and fuzzy demand assumptions [18] have been increasingly used with the aim to obtain more accurate and realistic results. In this paper, we deal with CLRP by using GA-ANN in this area.

LRP belongs to the class of NP-hard problems; CLRP is also NP-hard. Several mathematical models and exact solution procedures were developed for small and medium scale CLRPs in the literature [14-16]. Because of its complexity, CLRP requires the implementation of heuristic and meta-heuristic approaches [1]. While numerous exact methods can be effectively applied to small and medium scale problems, it is incapable of dealing with large-scale problems. Therefore, determination of the problem scale is very important in the first stage of the study.

## 3. CASE STUDY

The automotive company is located in the biggest urban settlement area in Turkey. The company has two active depots (D1 and D2), and they are shown as shadowed in Figure 1. In addition,  $D_i$  ( $i=3,\dots,9$ ) are shown as potential depots (determined by the



Figure 1 – The locations of depots and customers

automotive company) to be additionally evaluated. At first, the effectiveness of the current locations and routes is investigated, and the best possible depot(s) are searched to meet the growing demand for the next ten years. Customers (1,2,...30) are highlighted with stars and are un-uniformly spatially distributed across the area of 115,300 km<sup>2</sup>.

The stars denote the customers, and the circles denote the depots in *Figure 1*. The future customer demand is an important point to determine the depot location and the routings of the vehicles. Time series are applied to predict the future customer demand. Most of the time series (unemployment rate (UR), customer confidence index (CCI), inflation (I), industrial production index (IPI), economic confidence index (ECI), and automobile number (AN)) are gathered from the Turkish Statistical Institute. The customer demand is

obtained from the automotive company and depends on time. *UR*, *CCI*, *I*, *IPI*, *ECI*, and *AN* are inputs, and the total customer demand (*d*) data is the output for the GA-ANN model that will be explained later. The obtained time series cover the months between 2012 and 2016, providing 60 samples for training and testing in the prediction model. 20% of the instance is allocated for the testing, and the rest is allocated for the training stage in the prediction model. Let *UR(t)*, *CCI(t)*, *I(t)*, *IPI(t)*, *ECI(t)*, and *AN(t)* input variables be unemployment rate, customer confidence index, inflation, industrial production index, economic confidence index, and automobile number in time *t*, respectively. Here, *d(t)* is the total demand in time *t*. *Figure 2* shows each time series of input variables from 2012 to 2016. In addition, *Figure 3* shows time series for the total demand between 2012 and 2016.

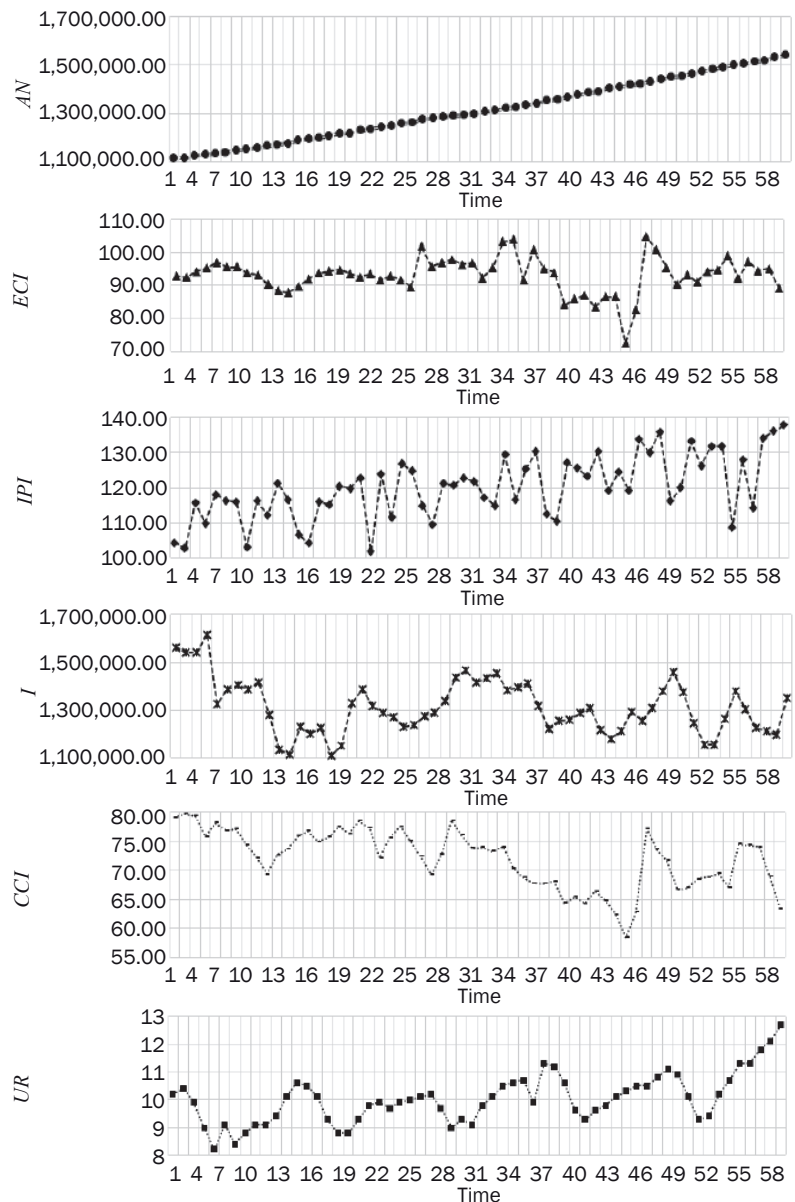


Figure 2 – All 6 time series

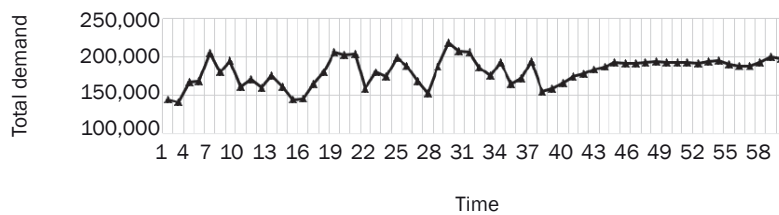


Figure 3 - Time series for  $d(t)$

#### 4. THE CONCEPTUAL FRAMEWORK

Since several different techniques are used in this paper, we are dealing with the multi-stage approach. The applied multi-stage approach is presented as a block diagram in Figure 4. Initially, a classification step is applied with k-NN in order to group the values of input variables. Later, the main effects of each input are calculated by using univariate analysis, and in the second step the significance level is considered to be 0.95. In the third step, the classified and analyzed input and output variables are used in GA-ANN to predict the total customer demand. In the fourth step, the VND algorithm is tested with a test instance from the literature and applied to determine depot locations and routings of vehicles with regard to total customer demand for ten years ahead, as predicted by using GA-ANN. In the last step, the final goal is achieved by lowering transportation costs.

#### 5. DEVELOPMENT OF THE DEMAND PREDICTING GA-ANN MODEL

This section describes the general background of the GA-ANN prediction model which will be implemented in the proposed system.

GA is initialized with an aggressive set of solution pools, and then the solution pools are improved through the stage of natural selection, where poor solutions run out, while the highest quality solutions survive to keep reproducing. This stage is iterated until the optimal condition is satisfied. A hybrid GA-based neural network is basically a backpropagation network, with the only exception being that the weight matrix is acquired from performing the genetic operations under optimal convergence conditions [6, 7].

The pseudo-code of the GA-ANN mechanism is illustrated in Algorithm 1. The initial weights  $w(iter)$  are set off randomly in the first iteration ( $iter=0$ ), and the output of each hidden neuron and the fitness function

$R^2$  are computed. Crossover procedure is applied in the reproduction stage  $w_r(iter)$ . Then, mutation process is applied to the new parents' weights  $w_p(iter)$ . The updated weights  $w_r(iter)$  are calculated with regard to GA by applying parent selection, reproduction, and mutation. The next iteration is carried out with regard to the updated weights. Various options, such as population, fitness scaling, selection, reproduction, mutation, crossover, and migration, need to be specified to tune the weights. The weights are tuned with respect to trial and error because there is no exact weight-tuning process.

Algorithm 1 - GA-ANN

- 1 Initialize initial weights ( $w(iter=0)$ )
- 2  $iter \leftarrow 0$ ;
- 3 Compute the output for every neuron
- 4 Compute the fitness function at the output
- 5 While not\_terminated() do
- 6  $w_p(iter) \leftarrow$  Select\_parents();
- 7  $w_r(iter) \leftarrow$  Reproduction();
- 8 Mutate  $w_r(iter)$ ;
- 9 Evaluate ( $w_r(iter)$ );
- 10  $w_r(iter) \leftarrow$  build\_next\_generation ( $w_r(iter)$ ,  $w(iter)$ )
- 11 End while
- 12  $iter \leftarrow iter+1$ ;

GA appropriately adjusts weights of the GA-ANN prediction model to forecast time-varying demand  $d$  based on time-varying inputs. The derived predicted demand is timely and quantitatively related to the actual demands of customers that appear in the CLRP problem. The prediction of the customer demand is carried out in two different time instances. Two different time instances are the years 2016 and 2026. The demand prediction for 2016 is carried out to improve the current performance of the transportation cost. The demand prediction for 2026 is carried out to determine the best depot location and the routings of vehicles.

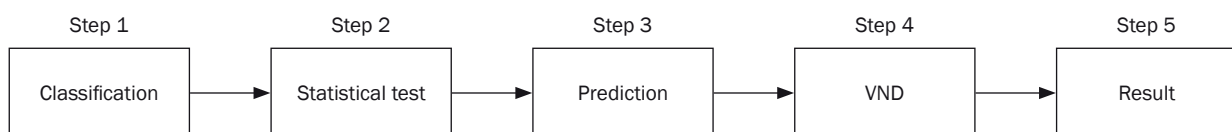


Figure 4 - The conceptual framework of the research

## 6. NUMERICAL RESULTS

### 6.1 Investigation of the GA-ANN inputs' main effects by using k-NN classification

Before the GA-ANN training stage, each input is classified with regard to the k-nearest neighbors (k-NN) to explore the main effects of each input in the GA-ANN prediction model with regard to the model's output *d*.

k-NN is one of the oldest and simplest non-parametric classification and regression methods [19]. In the k-NN algorithm, a class is allocated according to the most common class amongst its k-NN. The output is a class membership. The input consists of the *k* closest training examples. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its k-NN [5]. *UR*, *CCI*, *I*, *IPI*, *ECI*, and *AN* are classified with regard to demand *d* membership as shown in Figure 5. x-axis and y-axis show the customer demand and the value of the related input variable, respectively. Here,

*k* is a parameter that can influence the quality of the classification, and *k* is chosen as 3 to obtain better classification quality.

Table 1 presents the main effects of each input on the output demand that is calculated with regard to the 0.95 significance level. *UR*, *I*, *IPI*, and *AN* are highly significant in the prediction model because F test significance is higher than 0.05. However, *CCI* and *ECI* are not significant in the prediction model. Thus, GA-ANN is performed with regard to both variants: Variant 1. All variables (inputs); Variant 2. Inputs with an exclusion of *CCI* and *ECI*. The results of GA-ANN are presented in the following section.

### 6.2 Calibration of GA-ANN model structure

The proposed GA-ANN model is run with a combination of four different parameters, which are "population size", "crossover", "mutation", and "hidden neuron". Two population sizes are tested: 10 and 20. Ten different crossover values (0.1, 0.2,..., 1.0), ten different mutation values (0.1, 0.2,..., 1.0), and nineteen hidden neuron values (2, 3,..., 20) are tested. In total, 3,800(=2·10·10·19) different combinations (see

Table 1 – The main effects of each input of GA-ANN model with regard to the model's output *d*

Variable	Sum of squares of errors	Degree of freedom	Mean square of errors	F test	Significance
<i>UR</i>	936.541	4	234.135	3.431	0.017
<i>CCI</i>	230.296	3	76.765	1.125	0.351
<i>I</i>	952.114	5	190.423	2.791	0.030
<i>IPI</i>	2,856.460	3	952.153	13.954	0.000
<i>ECI</i>	9.630	2	4.815	0.071	0.932
<i>AN</i>	2,311.871	4	577.968	8.470	0.000
Error	2,592.865	38	68.233	-	-

Table 2 – Combinations tested for GA-based ANN

Population size	Crossover	Mutation	Hidden neuron
10, 20	0.1-1.0	0.1-1.0	2-20

Table 3 – The best 10 values of  $R^2$  with regard to six input variables

Population size	Crossover rate	Mutation rate	Hidden neuron	$R^2$ for training (mean value)	$R^2$ for testing (mean value)	Standard deviation for training	Standard deviation for testing
20	0.8	0.5	16	0.9526	0.8011	0.0087	0.0117
20	0.5	0.7	15	0.9509	0.7927	0.0071	0.0063
20	0.8	0.1	19	0.9508	0.7209	0.0103	0.0094
20	0.7	0.4	16	0.9503	0.7978	0.0060	0.0078
20	0.6	1	12	0.9496	0.7585	0.0116	0.0148
20	0.7	0.1	19	0.9494	0.7589	0.0234	0.0219
20	0.5	0.1	16	0.9492	0.7826	0.0249	0.0170
20	0.7	0.7	15	0.9490	0.7083	0.0161	0.0247
20	0.5	0.1	17	0.9473	0.7480	0.0122	0.0129
20	0.7	0.2	13	0.9469	0.7730	0.0140	0.0144

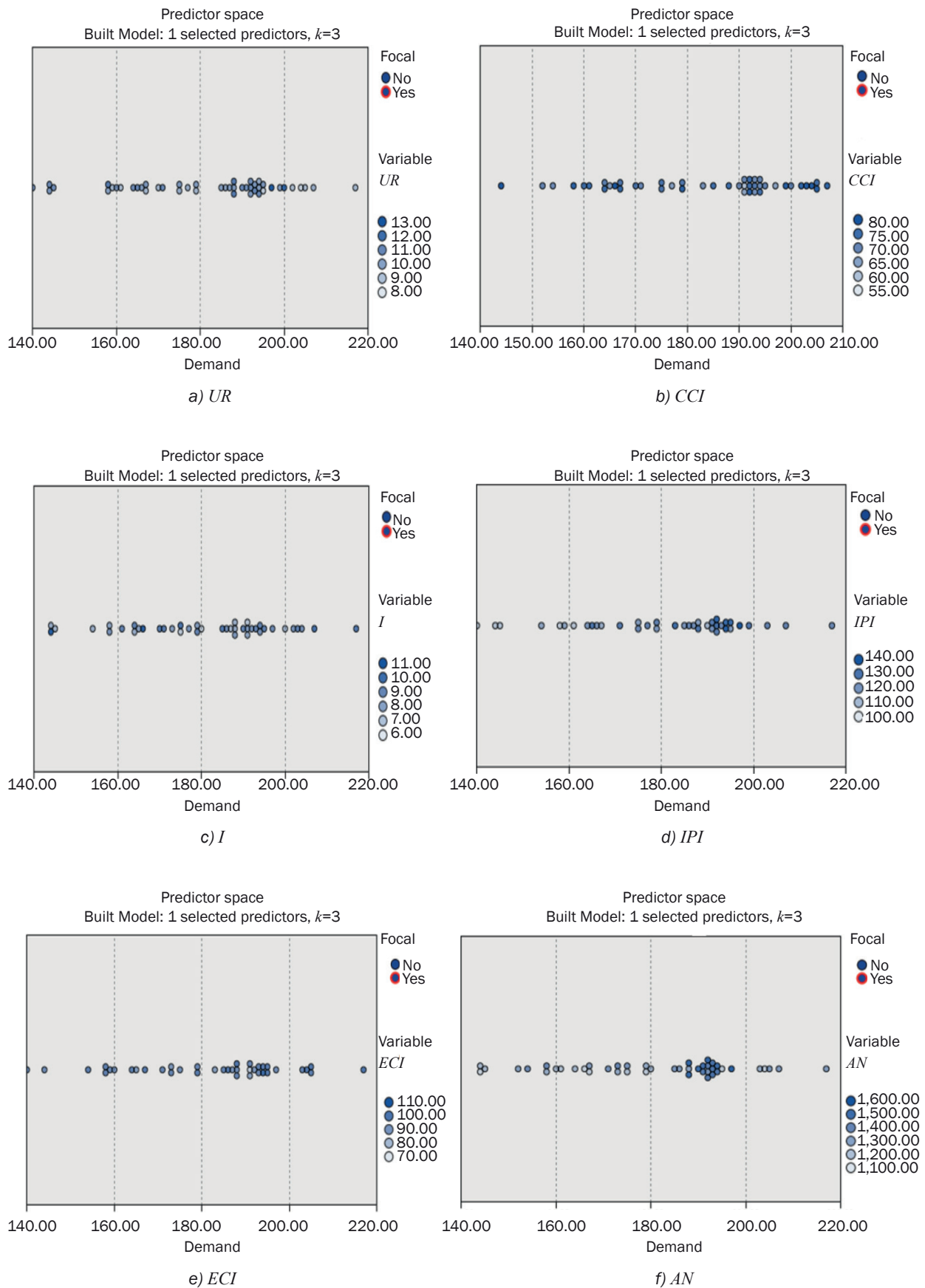


Figure 5 - The classification of time series

Table 4 – The best 10 values of  $R^2$  with regard to four input variables

Population size	Crossover rate	Mutation rate	Hidden neuron	$R^2$ for training (mean value)	$R^2$ for testing (mean value)	Standard deviation for training	Standard deviation for testing
20	0.7	1	20	0.7417	0.6607	0.0126	0.0132
20	0.4	0.4	17	0.7411	0.7056	0.0098	0.0081
20	0.8	0.7	17	0.7400	0.6805	0.0040	0.0272
20	0.8	1	16	0.7357	0.6593	0.0086	0.0301
20	0.6	0.8	12	0.7320	0.6499	0.0092	0.0252
20	0.8	0.5	7	0.7319	0.7191	0.0198	0.0296
20	0.5	0.8	7	0.7245	0.6564	0.0120	0.0092
20	0.7	0.1	14	0.7219	0.6256	0.0144	0.0134
20	0.7	0.6	7	0.7213	0.6949	0.0138	0.0186
10	0.5	0.1	14	0.7200	0.7039	0.0129	0.0152

Table 2) are tested to find the best (highest)  $R^2$  (determination coefficient) for training. Because of page limitation, we did not present all  $R^2$  values (except the best 10) for combinations.

The proposed GA-ANN is run on a computer that has a 32-bit Windows 7 operating system, 2.4-GHz processor, and 16-GB memory. GA-ANN is implemented in Matlab 7.12.

Table 3 shows the best 10 values of  $R^2$  with regard to six input variables (there is no exclusion – Variant 1). The best mean values ( $R^2$ ) of the training and testing stages are 0.9526 and 0.8011, respectively in Table 3.

As mentioned in Section 2, it was found that  $CCI$  and  $ECI$  are not significant in the prediction model. Thus,  $CCI$  and  $ECI$  were excluded from the GA-ANN model to see whether any improvement (based on  $R^2$ ) would happen or not. Table 4 shows the best 10 values of  $R^2$  with regard to the four input variables after the two variables ( $CCI$  and  $ECI$ ) were excluded from the model (Variant 2). However, on average, the exclusion of the two variables does not offer better  $R^2$  values than all parameters included. The best mean values ( $R^2$ ) of the training and the testing stages are 0.7417 and 0.6607, respectively, as presented in Table 4. However, these values are much lower than the best mean values in Table 3.  $R^2$  values are found as the result of the combination of population size, crossover rate, mutation rate, and hidden neuron number.

Transfer functions, which are used in the layer of GA-ANN, are logarithmic and purelin, respectively.

The optimal structure of GA-ANN weight-tuning that includes population, fitness scaling, selection, reproduction, mutation, crossover, and migration is shown in Table 5. The weights are tuned with respect to trial and error. As it turns out, the most appropriate population size, crossover fraction, migration fraction, and hidden neuron number are found to be 20, 0.80, 0.50, and 16, respectively. Gaussian mutation and heuristic crossover functions were applied in the mutation and crossover stage, respectively. GA-ANN

Table 5 – The optimal structures of GA-ANN weight-tuning

	Options	Optimal conditions
Population	Creation function Population size	20
Fitness scaling		-
Selection		Stochastic uniform
Reproduction	Elite count Crossover fraction	2 0.80
Mutation		Gaussian mutation
Crossover		Heuristic crossover
Migration	Direction Fraction	Both 0.50
Algorithm settings	Initial penalty Penalty factor	10 100
Stopping criteria	Generations Fitness limit Stall generations	1000 1.00e-8 50
Fitness function		$R^2$
Hidden neuron number	Single hidden layer	16
CPU time [s]		568

structure has some stopping criteria that include iteration number, fitness limit, and stall generations. Iteration number, fitness limit of fitness function  $R^2$ , and stall generation are set to 1,000, 1.00e-8, and 50, respectively. If there is no improvement in the course of 50 iterations, the searching process is stopped. In particular, when population size and hidden neuron increase, the computing time increases as well.

To sum up, a classification step is applied with k-NN in order to group the values of input variables. Later, the main effects of each input are calculated by using a univariate analysis, and the significance

level is considered to be 0.95 in the second step. In the third step, classified and analyzed input and output variables are used in GA-ANN to predict the future demand based on the values of input variables ( $CCI(2026)$ ,  $IPI(2026)$ ,  $ECI(2026)$ ,  $UR(2026)$ ,  $I(2026)$ ). These are measured by using three scenarios in the Equations 1-3 and the input variable ( $AN(2026)$ ) that is calculated with regard to the exponential model in Table 6.

### 6.3 Estimation of forecasted values for the year 2026

It is important to determine the values of the six input variables that are used in the demand  $d$  prediction stage, after the best combination of GA-ANN is found with regard to four different GA parameters such as population size, crossover, mutation, and hidden neuron. Three different categories of scenarios, which consist of the worst, the most probable, and the best, are constructed and calculated with regard to the data that cover the months between 2012 and 2016. The values of the input variables (except the  $AN$  variable) are measured via the three scenarios to predict the demand for 10 years ahead. Unlike the other input variables that are measured series (from 2012 to 2016), the  $AN$  input variable should be computed from a regression model for the period (2012–2016).

Table 6 shows the results of the four regression models for determining  $AN$ , which are linear (linear growth structure), logarithmic (logarithmic growth

structure), quadratic (parabolic growth structure), and exponential (exponential growth structure), in order to estimate the automobile number for ten years ahead. The exponential regression model offers the least residual and standard error. In addition, its significance value is lower than 0.05. Consequently, the exponential regression model was used to predict the variable  $AN$ .

Figure 6 shows the plot of the observed values and the estimated values  $AN(t)$  with regard to different regression models. The observed and the estimated values quite overlap in the exponential regression model. Figure 6 supports the results of the Table 6.

$AN(t)=\alpha \cdot e^{\beta \cdot t}$ ,  $t \in (2012-2016)$  is the equation of the exponential model that was involved in our forecasting. Here,  $\alpha$  and  $\beta$  are the estimated coefficients amounting to 1,103.276 and 0.006, respectively.

Three different scenarios are performed to determine the customer demand, especially at two extreme points (the worst and the best). The worst, the most probable, and the best values of each of the five ( $CCI(2026)$ ,  $IPI(2026)$ ,  $ECI(2026)$ ,  $UR(2026)$ ,  $I(2026)$ ) input variables are calculated with regard to the following formulas in the Equations 1-3. Here,  $IV(t)$  denotes one of the five input variables in time  $t$ . Equation 1 calculates the worst values of the input variables. The worst values imply the maximum value for  $UR$  and  $I$  and the minimum value for  $CCI$ ,  $IPI$ , and  $ECI$  between the years 2012 and 2016. Equation 2 calculates the most probable values of the input variables considering the mean values of input variables. Equation 3

Table 6 – The results of four regression models for  $AN(t)$ ,  $t=2026$

Model	Structure	R	R <sup>2</sup>	Std. error of the estimate	Residual	Sig.
Linear	$\alpha + \beta \cdot t$	0.999	0.997	6.530	2473.484	0.000
Logarithmic	$\alpha + \beta \cdot \ln(t)$	0.887	0.787	58.942	201,498.533	0.000
Quadratic	$\alpha + \beta \cdot t + \gamma \cdot t^2$	1	1	2.400	328.305	0.000
Exponential	$\alpha \cdot e^{\beta \cdot t}$	1	1	0.002	0.000	0.000

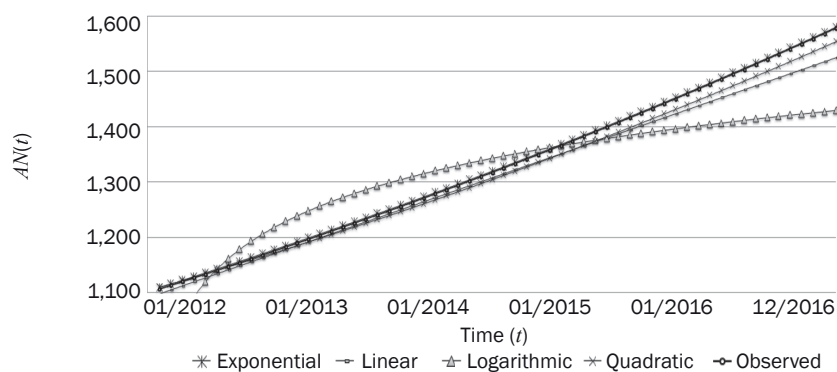


Figure 6 – The plot between the observed values and the estimated values with regard to regression models for  $AN(t)$ ,  $t=2012, \dots, 2016(x1,000)$



calculates the best values of input variables. The best values imply the maximum values for *CCI*, *IPI*, and *ECI* and the minimum value for *UR* and *I* between the years 2012 and 2016.

$$IV(2026) = (\max(IV(t)) \forall IV \in UR, I, \min \forall (IV(t)) \forall IV \in CCI, IPI, ECI), \quad (1)$$

$$t = 1, \dots, 60$$

$$IV(2026) = \frac{\sum_{t=1}^{t=60} IV(t)}{\text{total sample number}} \forall IV \in UR, CCI, I, IPI, ECI, \quad (2)$$

$$t = 1, \dots, 60$$

$$IV(2026) = (\max(IV(t)) \forall IV \in CCI, IPI, ECI; \min(IV(t)) \forall IV \in UR, I) \quad (3)$$

$$t = 1, \dots, 60$$

*CCI* (2026), *IPI* (2026), *ECI* (2026), *UR* (2026), and *I* (2026) input variables are measured with regard to the best, the most probable, and the worst values of the input variables by considering the data that is obtained for the period 2012–2016, while the *AN* (2026) input variable is calculated by the exponential regression model.

Table 7 shows the worst, the most probable, and the best values of the input variables obtained to predict the customer demand for the year 2026. The total demand *d*(2026), calculated with regard to the values of input variables for the year 2026, is later used in the CLRP model. The total demand *d*(2026) consists of the demand of thirty customers. The prediction procedure is applied to the demand of each customer (*d<sub>i</sub>*) and the sum of the predicted customer demand is presented as the total demand. The total demand values *d*(2026) of the best, the most probable, and the worst scenarios amount to 445,000, 418,000, and 347,000 units, respectively. These values are derived by using the forecasts based on inputs for the year 2026 in Table 7, which are injected into the GA-ANN model to calculate the output *d*(2026) for all three scenarios.

The automobile number *AN*(*t*) is estimated to be 3,248,794 for ten years ahead (for the year 2026), and the forecasts for each input variable for the year 2026 are shown in Table 7.

Table 7 – The worst, the most probable, and the best values of the input variables ten years ahead

Input variable	The worst	The most probable	The best
<i>UR</i> (2026)	12.70	10	8.70
<i>CCI</i> (2026)	58.50	71.81	79.65
<i>I</i> (2026)	11.14	8.20	6.13
<i>IPI</i> (2026)	102.10	120.15	137.90
<i>ECI</i> (2026)	72.60	92.96	104.80
<i>AN</i> (2026)	-	3,248,794	-

## 7. CLRP MODEL

CLRP deals with the placement of facility location and routing between the customer and facilities or depots simultaneously by considering that the capacities of depots and vehicles are not violated. In our case study, it is assumed that vehicles are shared by all depots and all vehicles are homogeneous. The following constraints must be satisfied [8]:

- 1) The total demand of customers assigned to one depot must not exceed its capacity
- 2) Each route starts and ends at the same depot
- 3) Each vehicle is assigned to one trip at most
- 4) Each customer is allocated to one single vehicle and one single depot
- 5) The total demand of customers assigned to one vehicle must not exceed its capacity

Tables 8 and 9 present the parameters and decision variables of CLRP, respectively. The descriptions of the parameters and decision variables are presented in detail.

The descriptions of all equations are presented; later all equations are introduced.

$$\min \sum_{i \in N} \sum_{j \in N} c_{ij}x_{ij} + \sum_{k \in N_d} O_k y_k + \sum_{k \in N_d} \sum_{i \in N_c} VCx_{ki} \quad (4)$$

$$s.t. \sum_{j \in N} x_{ij} = 1 \quad \forall i \in N_c \quad (5)$$

$$\sum_{j \in N} x_{ji} = \sum_{j \in N} x_{ij} \quad \forall i \in N \quad (6)$$

$$\sum_{k \in N_d} z_{ik} = 1 \quad \forall i \in N_c \quad (7)$$

$$x_{ik} \leq z_{ik} \quad \forall i \in N_c, \forall k \in N_d \quad (8)$$

Table 8 – Parameters of CLRP

Notation	Description
<i>N<sub>c</sub></i>	Set of customers
<i>N<sub>d</sub></i>	Set of depots
<i>N</i>	All set
<i>c<sub>ij</sub></i>	Transportation cost between two nodes
<i>d<sub>i</sub></i>	Demand for <i>i</i> -th customer
<i>AK</i>	Vehicle capacity
<i>VC</i>	Vehicle fixed cost (for each route)
<i>dk<sub>k</sub></i>	Capacity of <i>k</i> -th depot
<i>O<sub>k</sub></i>	Opening cost of <i>k</i> -th depot

Table 9 – Decision variables of CLRP

Notation	Description
<i>x<sub>ij</sub></i>	A vehicle travels from node <i>i</i> to node <i>j</i> or not
<i>z<sub>ik</sub></i>	<i>i</i> -th customer is assigned to <i>k</i> -th depot or not
<i>y<sub>k</sub></i>	<i>k</i> -th depot is opened or closed
<i>u<sub>ij</sub></i>	Delivery load on the vehicle from node <i>i</i> to node <i>j</i>

$$x_{ki} \leq z_{ik} \quad \forall i \in N_c, \forall k \in N_d \quad (9)$$

$$x_{ij} + z_{ik} + \sum_{m \in N_d, m \neq k} z_{jm} \leq 2 \quad \forall i, j \in N_c, i \neq j, \forall k \in N_d \quad (10)$$

$$\sum_{k \in N_d} d_i z_{ik} \leq dk_k y_k \quad \forall k \in N_d \quad (11)$$

$$\sum_{j \in N} u_{ji} - \sum_{j \in N} u_{ij} = d_i \quad \forall i \in N_c \quad (12)$$

$$\sum_{j \in N_c} u_{ij} = \sum_{j \in N_c} z_{jk} d_j \quad \forall k \in N_d \quad (13)$$

$$\sum_{j \in N_c} u_{jk} = 0 \quad \forall k \in N_d \quad (14)$$

$$u_{ij} \leq (AK - d_i)x_{ij} \quad \forall i \in N_c, \forall j \in N \quad (15)$$

$$u_{ij} \geq d_j x_{ij} \quad \forall i \in N, \forall j \in N_c \quad (16)$$

$$x_{ij} \in \{0, 1\} \quad \forall i, j \in N \quad (17)$$

$$z_{ik} \in \{0, 1\} \quad \forall i \in N_c, \forall k \in N_d \quad (18)$$

$$y_k \in \{0, 1\} \quad \forall k \in N_d \quad (19)$$

Equation 4 is the objective function that minimizes the total system cost, including transportation, depot, and vehicle fixed costs. Equation 5 ensures that each customer must be visited exactly once, Equation 6 guarantees that the number of entering and leaving arcs to each node is equal. Equation 7 ensures that each customer must be assigned to only one depot. Equations 8-10 avoid the illegal routes, which do not start and end at the same depot. Equation 11 guarantees that the total delivery on any depot does not exceed the corresponding depot capacity. Equation 12 is flow conservation constraints for delivery demands. Equation 13 ensures that the total delivery load dispatched from each opened depot equals to the total delivery demand of customers, which are assigned to the corresponding depot. Equation 14 guarantees that the total delivery load returning to the opened depots must be equal to zero. Equations 15 and 16 are bounding constraints for the additional variables. Equations 17-19 show that  $x_{ij}$ ,  $z_{ik}$ , and  $y_k$  are the binary variables.

## 8. VARIABLE NEIGHBORHOOD SEARCH FOR THE CLRP

Various methods have been applied in LRP. Variable neighborhood search (VNS) is a metaheuristic method for solving several difficult problems. Consider a combinatorial problem to minimize a function  $f$  defined on a solution space  $X$ .  $x \in X$  is related to a subset  $\mathfrak{x}(x) \subseteq X$  called the neighborhood of  $x$ . Let  $\mathfrak{x}_l$ ,  $l \in \{1, \dots, l_{max}\}$  be the set of the neighborhood structures chosen to be explored during the search.  $\mathfrak{x}_l(x)$  is the set of solutions of the  $l$ -th neighborhood of  $x$  [9]. VNS has the three basic facts that assure the achievement of neighborhoods, and they are stated as follows [10, 11]:

- 1) A local minimum with respect to one neighborhood structure is not necessarily the same for another
- 2) A global minimum is a local minimum with respect to all possible neighborhood structures
- 3) For many problems, local minima with respect to one or several neighborhoods are relatively close to each other

### 8.1 Representation of the LRP algorithm

The effectiveness of the algorithm strictly depends on the solution encoding. In our implementation, let  $s$  be a feasible solution, consisting of a set of sequences  $r_b$  for  $b=1, \dots, m$ . Consider the case of three possible locations of depots and 9 customers. Figure 7 shows a representative solution of an LRP. Customers 3 and 5 are assigned to depot D1, customers 8, 6, 9, and 2 are assigned to depot D3, and customers 1, 4, and 7 are assigned to depot D5. The initial and the last visited nodes are assigned to the related depot in their route.

### 8.2 Variable neighborhood descent (VND)

In this paper, we applied the VND method with a slight modification to the first neighborhood structure  $N_1$ . The first neighborhood structure is applied by considering the second neighborhood structure simultaneously.

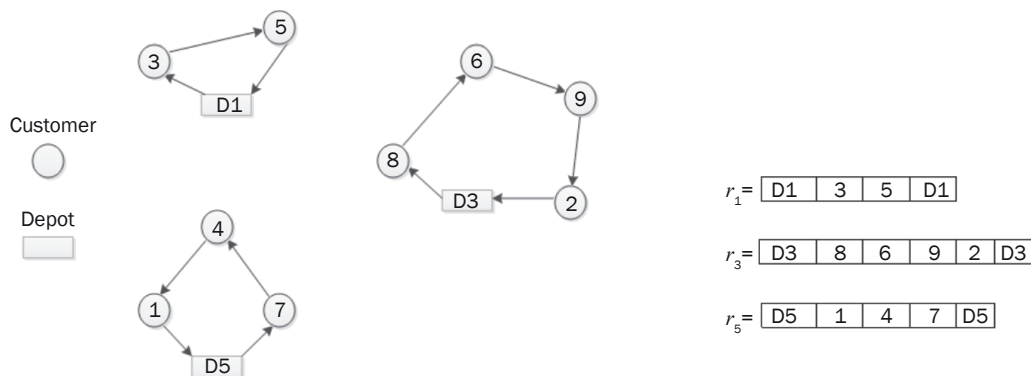


Figure 7 - Presentation of the LRP solution on the representative example

The VND method [11] starts by choosing an initial solution  $x$  and searches a direction of descent from  $x$  within a neighborhood structure. If the direction of descent is present, the heuristic is iterated, and otherwise it stops. In this study, the VND algorithm consists of five neighborhood structures ( $N_1, N_2, N_3, N_4$  and  $N_5$ ) to search distinct possibilities of depot locations and to improve the assignment of customers to each depot. *Figure 8* shows the first neighborhood  $N_1$ . It performs that a customer, the first or the last node in the route, is removed from the route and added to a customer, the first or the last node in different or same route. The first neighborhood structure is applied to each customer pair (the first and the last node) by considering the second neighborhood structure simultaneously. The second neighborhood  $N_2$  performs that

one customer is shifted from its current position to another position, in the same route or in a different route which may be assigned to the same depot or the different depot (considering the assigned depot capacity) [9]. *Figure 9* shows an example of  $N_2$ . The third neighborhood  $N_3$  performs that two customers, who may belong to the same route or to two distinct routes sharing one common depot or not, are exchanged, if residual capacities allow it [9]. *Figure 10* shows an example of  $N_3$ . The fourth neighborhood  $N_4$  performs the removal of two non-consecutive edges, either in the same route or in two distinct routes assigned to a common depot or not. When they belong to different routes, the reconnection of the trips can be performed in a number of ways. If they are from different depots, edges connecting the last customers of the two considered

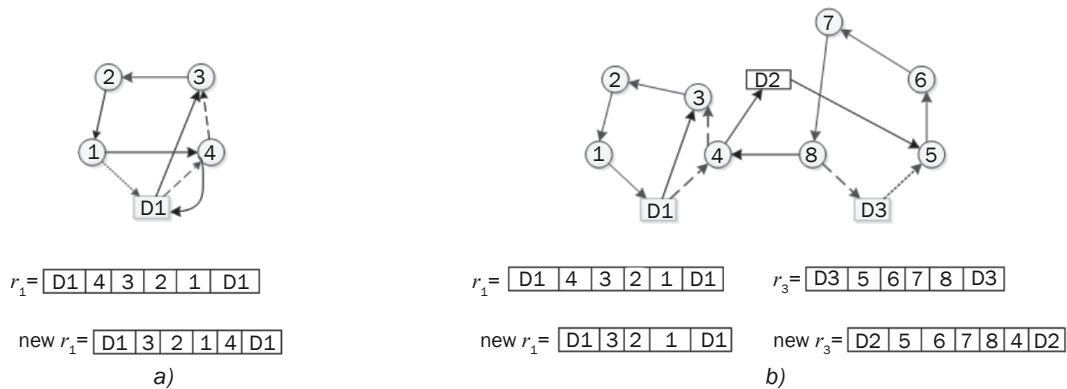


Figure 8 - An example of  $N_1$

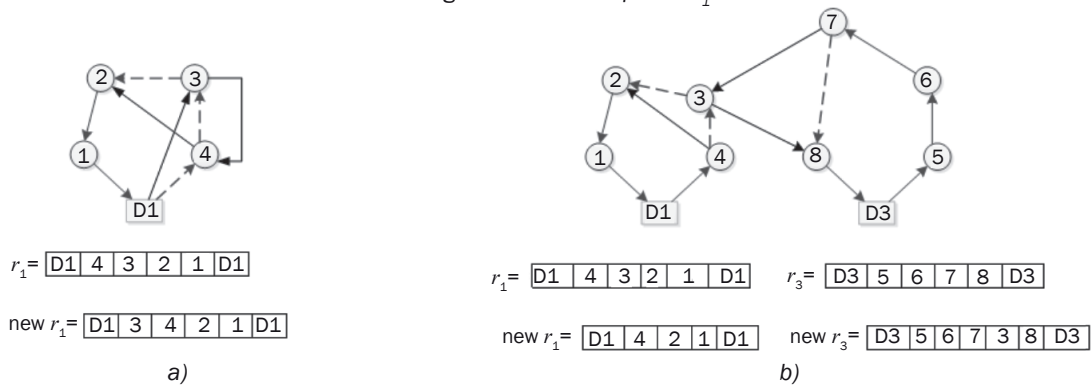


Figure 9 - An example of  $N_2$ -insertion

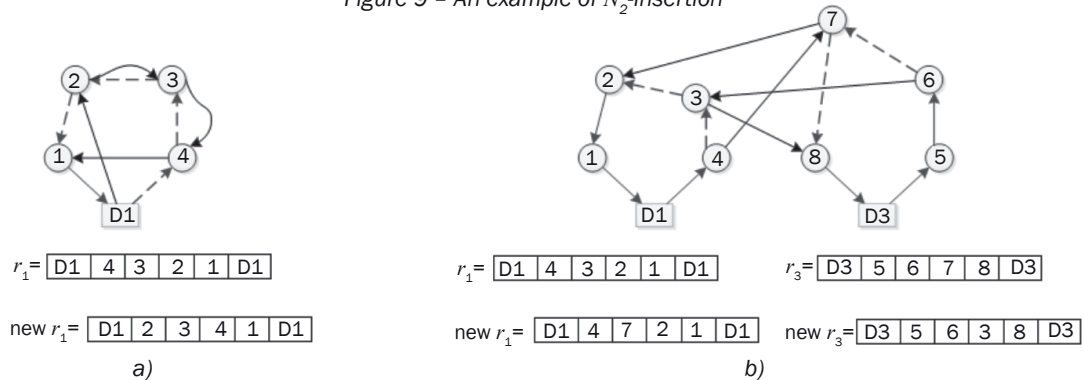


Figure 10 - An example of  $N_3$ -swap

routes to their depot have to be replaced to satisfy the constraint imposing that a route must start and end at the same depot [9]. Figure 11 shows an example of  $N_4$ . The fifth neighborhood  $N_5$  performs the insertion of a sequence of customers. A set of nodes is chosen from one route and inserted to a feasible route [9]. Figure 12 shows an example of  $N_5$ . The VND algorithm chooses neighborhoods with regard to the sequential deterministic order. The neighborhood structures are searched one by one in the given order.

An evaluation function has an important role in reaching the desired result in the VND algorithm. A generated route  $r_b$  of a solution  $x$  begins from  $r_b^0$ , moves through a sequence of  $n_b$  customers, and returns to the same depot  $r_b^{n_b+1}$ . Let  $q(r_b)$  be the total load of route  $r_b$ . The cost of the generated route is shown in Equation 20. The penalty for exceeding the depot capacity is represented by  $\theta_b$  [11].

$$c(r_b) = \theta_b + \sum_{i=0}^{n_b} c_{r_b, r_b^{i+1}} + \theta_b \max(0, q(r_b) - dk_b) \quad (20)$$

The evaluation function is performed in each neighborhood structure ( $N_l$ ) in order to avoid exceeding the capacity constraint of the depots and vehicles and minimize the transportation cost. The pseudo code of the performed VND is illustrated in Algorithm 2.

An important phase that is applied before the VND algorithm starts is the shaking procedure. It generates a point  $x'(x' \in N_l(x))$  at random from the  $l$ -th neighborhood of  $x$ . The shaking phase aims to offer a beneficial starting point by diversifying the searching space for the local search. In this paper, the shaking is applied at random with regard to  $N_1, N_2, N_3, N_4$ , and  $N_5$  before the VND algorithm is applied in Algorithm 2. The shaking alternates between  $N_1$  and  $N_5$  at each iteration by starting with one of them with regard to a given probability that is defined in the parameter settings.

Algorithm 2 - The performed VND

- 1 **Input:** The set of neighborhood structures  $N_l$  for  $l=1, \dots, l_{max}=4$
- 2 **Initialization:**  $l \leftarrow 1$ ; find a random initial solution  $x$
- 3 **Shaking**
- 4 **while**  $l \leq l_{max}$  **do**
- 5 **if**  $x$  is not a local optimum for the neighborhood structure  $N_l$  **then**
- 6  $x' \leftarrow$  the first solution in  $N_l(x)$
- 7  $x \leftarrow x'; l \leftarrow 1$ ;
- 8 **else**  $l \leftarrow l+1$ ;
- 9 **return**  $x$ ;

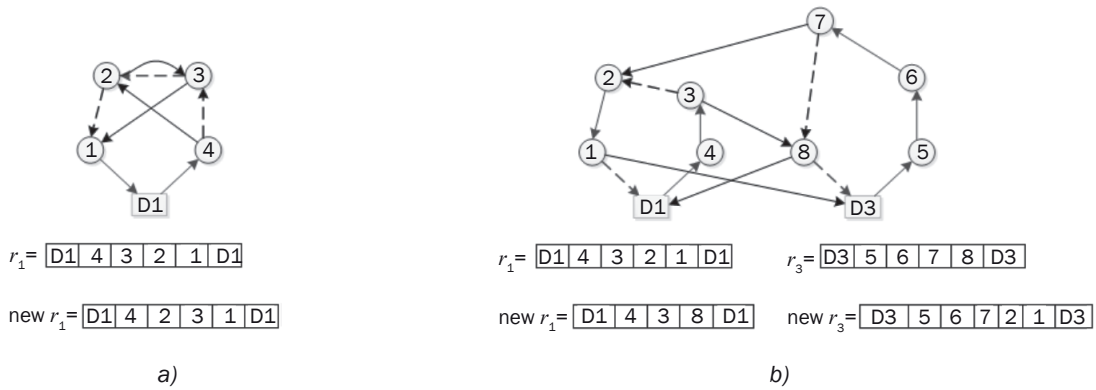


Figure 11 – An example of  $N_4$ -two opt

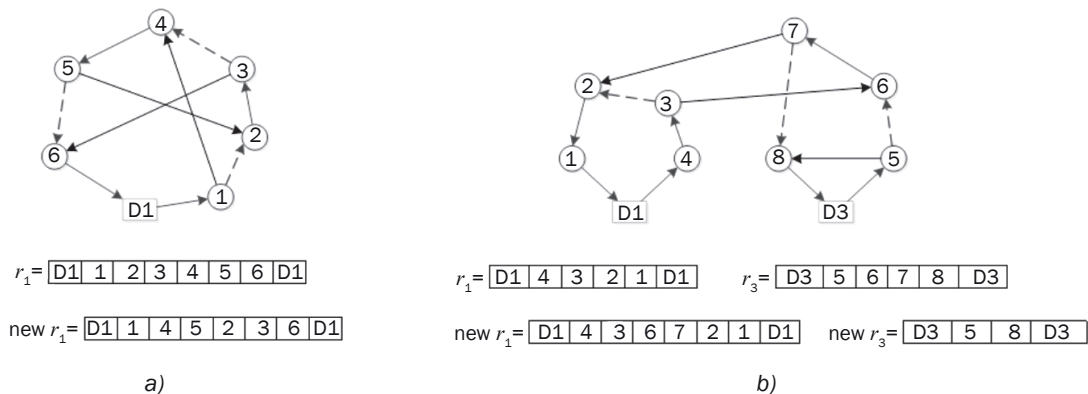


Figure 12 – An example of  $N_5$

## 9. RESULTS AND DISCUSSION

### 9.1 Comparison of the performed VND algorithm with Prins’s instances

The performed algorithm is implemented in C# language using a laptop with the Intel i7, 2.4 GHz processor with 16 GB memory. The efficiency of the performed algorithm is illustrated by using a number of test instances (Prins et al.’s instances [8]) in Table 10. Table 10 shows the comparison between the best known solutions and the results that are obtained via VND. First four instances consist of 20 customers and 5 depots, and the applied VND algorithm found the optimal solutions for these test instances. The rest of the test instances consists of 50 customers and 5 depots, and the applied VND algorithm reaches the solutions with the average gap of 0.67%. The test instances, with up to 50 customers, are performed in order to present the performed algorithm’s efficiency because the company has 30 customers and seven potential depot locations.

### 9.2 The results of the VND algorithm applied to the current year

Firstly, the VND algorithm is applied to the current year in order to improve routings of vehicles. Figure 13 shows the existing depots and routings in the company. The company has two existing depots (D1 and D2) and nine different vehicle fleets. The transportation cost of existing depots and routings of the vehicles is \$274,740. The existing depots and routings of the vehicles are shown to indicate the difference between the existing and the proposed system’s depots and routings of the vehicles.

Figure 14 shows the improved routings in the company. The company still has two opened depots (D1 and D2) and nine different vehicle fleets in the

proposed method. The cost of the proposed depots and routings is \$268,270. The current performance of the transportation cost is improved by 2.35%. This improvement rate offers a fair saving for the company.

If depots D1 and D2 had not been opened in the first construction stage, the company would have set up three depots (D3, D4, and D6) and eight different vehicle fleets with regard to the applied VND. This condition is illustrated in Figure 15. The cost of the depots and routings would have amounted to \$212,550.

Route 1	D1	4	9	D1			
Route 2	D1	5	6	D1			
Route 3	D1	15	14	13	10	D1	
Route 4	D1	25	7	29	D1		
Route 5	D1	30	3	D1			
Route 6	D1	2	8	20	1	D2	
Route 7	D1	11	21	12	23	D2	
Route 8	D1	17	28	22	26	24	D2
Route 9	D1	18	19	27	16	D2	

Figure 13 – Current performed routings

Route 1	D1	4	D1				
Route 2	D1	9	15	14	13	D1	
Route 3	D1	16	18	19	27	10	D1
Route 4	D1	24	17	28	22	26	D1
Route 5	D1	29	7	25	D1		
Route 6	D1	2	8	20	1	D2	
Route 7	D1	5	6	D2			
Route 8	D1	11	12	23	21	D2	
Route 9	D1	30	3	D2			

Figure 14 – Improved routings

Route 1	D3	3	2	8	20	D3	
Route 2	D3	4	5	D3			
Route 3	D3	6	30	D3			
Route 4	D4	10	21	11	1	15	D4
Route 5	D4	12	23	13	14	D4	
Route 6	D4	16	18	19	27	D4	
Route 7	D6	17	29	24	26	22	D6
Route 8	D6	28	7	9	25	D6	

Figure 15 – The routings without D1 and D2

Table 10 – Solutions on Prins et al.’s instances

Instance	Best known solution	Solution of the algorithm	CPU [s]	% gap
20-5-1a	54,793	54,793	1.20	0.00
20-5-1b	39,104	39,104	0.50	0.00
20-5-2a	48,908	48,908	1.24	0.00
20-5-2b	37,542	37,542	0.42	0.00
50-5-1	90,111	90,908	23.21	0.89
50-5-1b	63,242	64,761	31.45	2.40
50-5-2	88,298	88,786	21.59	0.55
50-5-2b	67,308	68,155	20.16	1.26
50-5-2bis	84,055	85,115	21.42	1.26
50-5-2bbis	51,822	51,902	25.81	0.15
50-5-3a	86,203	87,380	18.05	1.37
50-5-3b	61,830	61,890	21.46	0.10
Average	64,434.67	64,937	15.54	0.67

If D1 and D2 had not been opened in the first construction stage, the cost of current routings, which are performed in the existing system, would have been improved by 22.63% with regard to the applied VND. This improvement rate offers a reasonable saving for the company.

### 9.3 The results of VND algorithm applied to ten years ahead

We applied a multi-stage approach, which mainly consisted of classification, GA-ANN prediction, and the application of VND algorithm to improve the routings for the current routings and to determine optimal depot locations and routings of vehicles by evaluating the best and the worst cases of demand quantity for ten years ahead. Now, *Table 11* shows the cost of three different cases (the worst, the most probable, and the best) for ten years ahead. Each case is experimented with nine different fleet sizes, and the results of 27 different scenarios are shown in *Table 11*. The customers and the depots that are determined in *Table 11* are illustrated in *Figure 1*.

The best fleet size of the worst case is found to be 1,000, the cost is calculated as \$314,660- Depots D1, D2, and D8 are decided to be opened, and the total number of required routes is four for ten years ahead. Since the demand in the worst case is lower than in the other cases, it requires the lowest vehicle number.

Similar to the worst case, the best fleet size of the best case is found to be 1,000, the cost is calculated as \$381,260. Depots D1, D2, D3, and D8 are decided to be opened, and the total number of required routes is five for ten years ahead. Since the demand in the best case is higher than in the worst case, it requires a higher vehicle number.

Unlike the worst and the best cases, the best fleet size of the most probable case is found to be 800, the cost is calculated as \$375,370. Depots D1, D2, D3, and D6 are decided to be opened, and the total number of required routes is six for ten years ahead.

The capacity of each vehicle is 100 units, and if the fleet size equals 300 units, the required vehicle number is 3 per route. The current routing system has 9 routes, and each route has a 400 fleet size, so the company requires 36 (4x9) vehicles. However, with the growing customer demand for the year 2026, the required vehicle number of the worst, the most probable, and the best cases will be 40 (10x4), 48 (8x6), and 50 (10x5), respectively.

To sum up, the company already uses D1 and D2, but new depots are required to meet the growing demand. The results show that the company will use four or five different routes and will need to open at least one depot. New depots to be opened for ten years ahead are D3-D6 and D3-D8 for the most probable and the best cases, respectively. On the other hand, the new depot to be opened for ten years ahead is D8 for the worst case.

## 10. CONCLUSION

This paper aims to find the optimal depot locations and vehicle routings for spare parts of an automotive company considering the future demands. We applied a novel multi-stage approach to improve the current vehicle routings and determine new depot location(s) and vehicle routings for ten years ahead. At first, each factor was classified with regard to k-NN in order to generate variable levels. The factors that affect the customer demand were investigated and tested using a univariate analysis. The performed univariate analysis found that CCI and ECI were not significant in the prediction model. However, the exclusion of the two parameters did not offer better  $R^2$  values than other parameters included. Therefore, all parameters were included in the prediction model. Later, GA and ANN were hybridized and applied to predict the customer demand by considering all factors for the current year and ten years ahead.

After GA-ANN was applied successfully, the VND method was used to determine the location of depots and routings of the customers for the current year. The VND method was carried out with a slight modification

*Table 11 – The cost of the three different cases for ten years ahead (year 2026)*

	The worst			The most probable			The best		
	Cost (\$)	Depot	Route	Cost (\$)	Depot	Route	Cost (\$)	Depot	Route
Fleet size: 300	361,500	1, 2, 6, 8	14	412,330	1, 2, 3, 5	17	431,990	1, 2, 3, 7, 8	17
Fleet size: 400	355,690	1, 2, 6, 8	10	402,120	1, 2, 3, 4	13	435,010	1, 2, 3, 5, 8	13
Fleet size: 500	345,620	1, 2, 4	9	391,680	1, 2, 3, 4	9	426,080	1, 2, 3, 6, 8	10
Fleet size: 600	319,450	1, 2, 3	6	377,130	1, 2, 3, 8	8	416,210	1, 2, 5, 6, 8	8
Fleet size: 700	327,070	1, 2, 3	6	376,760	1, 2, 3, 8	7	410,320	1, 2, 7, 8, 9	7
Fleet size: 800	324,650	1, 2, 3	5	375,370	1, 2, 3, 6	6	396,870	1, 2, 3, 9	7
Fleet size: 900	317,760	1, 2, 3	4	382,490	1, 2, 3, 8	6	382,220	1, 2, 6, 8	6
Fleet size: 1,000	314,660	1, 2, 8	4	379,220	1, 2, 3, 9	5	381,260	1, 2, 3, 8	5
Fleet size: 1,100	321,580	1, 2, 8	4	384,340	1, 2, 6, 8	5	384,540	1, 2, 3, 8	5

to the first neighborhood structure. The first neighborhood structure was applied by considering the second neighborhood structure simultaneously. The company has two existing depots (D1 and D2) and nine different vehicle fleets in the current operating system. The transportation cost of existing depots and routings is \$274,740. The transportation cost of the proposed design for depots and vehicle routings is \$268,270. Therefore, the performed VND method offers 2.35% improvement. This improvement rate offers a fair saving for the company.

On the other hand, if depots D1 and D2 had not been opened in the first construction stage, the company would have set up three depots (D3, D4, and D6) and eight different vehicle fleets. The transportation cost of depots and routings of vehicles would have amounted to \$212,550. This case offers a 22.63% improvement.

Finally, the best, the most probable, and the worst cases were experimented with 27 different scenarios which had nine different fleet sizes. The results show that a new depot or more of them are required to meet the growing demand with regard to the performed VND algorithm for ten years ahead. In addition, the result illustrates that the company should use 4 or 5 different routes and open at least one more depot to satisfy future demands. New depots to be opened are D3, D6 for the most probable case and D3, D8 for the best case, respectively. However, in the worst case the choice is depot D8.

For future research, a robust optimization model can be created using the VND algorithm, and both results may be compared.

**ENGİN PEKEL**, Ph.D. Adayı,<sup>1</sup>

E-mail: enginpekel@hitit.edu.tr, pekkel@hotmail.com

**SELİN SONER KARA**, Ph.D.<sup>2</sup>

E-mail: ssoner@yildiz.edu.tr

<sup>1</sup> Mühendislik Fakültesi, Endüstri Mühendisliği Bölümü,  
Hitit Üniversitesi, 19030, Çorum, Türkiye

<sup>2</sup> Makina Fakültesi, Endüstri Mühendisliği Bölümü,  
Yıldız Teknik Üniversitesi,  
A-Blok, 34300, Beşiktaş, İstanbul, Türkiye

## **ALÇALAN DEĞİŞKEN KOMŞULUK VE GENETİK ALGORİTMA – YAPAY SİNİR AĞLARI HİBRİT METODU İLE KAPASİTE KISITLI YER SEÇİMİ VE ROTALAMA PROBLEMİNİN ÇÖZÜMÜ**

### **ÖZET**

*Bu çalışma, bir otomotiv yedek parça firması için gelecek talepleri göz önüne alarak optimal depo yerlerini ve araç rotalarını bulmayı amaçlamaktadır. Çeşitli metotların uygulandığı Kapasite Kısıtlı Yer Seçimi ve Rotalama (KKYSR) problemi Yapay Sinir Ağlarını (YSA) kullanarak, optimal depo yerlerini ve araç rotalarını bulmak için uygulanmıştır. Daha düşük ulaştırma maliyeti sağlayan, yeni çok aşamalı bir yaklaşım, KKYSR probleminde yürütülmüştür. İlk olarak,*

*müşteri talebi için önemli olan faktörler tek değişkenli analiz ile test edildikten sonra, girdiler tahmin aşamasında kullanılmıştır. Daha sonra, Genetik Algoritma (GA) ve YSA hibritlenmiş ve gelecek talep verisini sağlamak için uygulanmıştır. Depo yerleri ve araç rotaları Alçalan Değişken Komşuluk (ADK) algoritması kullanılarak belirlenmiştir. Rotalama ya da yer tipinin olduğu 5 komşuluk yapısı hem sallama hem de yerel arama adımıyla uygulanmıştır. GA-ANN ve ADK ilgili adımlarda başarı ile yürütülmüştür. Uygulanan ADK algoritması sayesinde, firma mevcut yıl için ulaştırma maliyetini %2.35 düşürmekte ve 10 yıl sonrası için en iyi ve en kötü müşteri talebi durumlarını değerlendirerek optimal depo yerlerini belirlenmesi ve araçların rotalanması imkanı vermektedir.*

### **ANAHTAR KELİMELELER**

*Alçalan değişken komşuluk; Genetik algoritma; Kapasite kısıtlı yer seçimi ve rotalama problemi; K-en yakın komşuluk; Sezgiseller; Yapay sinir Ağları;*

### **REFERENCES**

- [1] Karaoglan I, Altıparmak F. A memetic algorithm for the capacitated location-routing problem with mixed backhauls. *Computers & Operations Research*. 2015;55: 200-216.
- [2] Drexel M. Applications of the vehicle routing problem with trailers and transshipments. *European Journal of Operational Research*. 2013;227(2): 275-283.
- [3] Duhamel C, Lacomme P, Prins C, Prodhon C. A GRASP×ELS approach for the capacitated location-routing problem. *Computers & Operations Research*. 2010;37(11): 1912-1923.
- [4] Escobar JW. Heuristic algorithms for the capacitated location-routing problem and the multi-depot vehicle routing problem. *4OR*. 2014;12(1): 99.
- [5] Chen HL, Yang B, Wang G, Liu J, Xu X, Wang SJ, Liu DY. A novel bankruptcy prediction model based on an adaptive fuzzy k-nearest neighbor method. *Knowledge-Based Systems*. 2011;24(8): 1348-1359.
- [6] Kadiyala A, Kaur D, Kumar A. Development of hybrid genetic-algorithm-based neural networks using regression trees for modeling air quality inside a public transportation bus. *Journal of the Air & Waste Management Association*. 2013;63(2): 205-218.
- [7] Pekel E, Soner Kara S. Passenger Flow Prediction Based on Newly Adopted Algorithms. *Applied Artificial Intelligence*. 2017;31(1): 64-79.
- [8] Prodhon C, Prins C. A survey of recent research on location-routing problems. *European Journal of Operational Research*. 2014;238(1): 1-17.
- [9] Jarboui B, Derbel H, Hanafi S, Mladenović N. Variable neighborhood search for location routing. *Computers & Operations Research*. 2013;40(1): 47-57.
- [10] Hansen P, Mladenović N. *Variable neighborhood search. Handbook of metaheuristics*. Boston, Dordrecht, London: Kluwer Academic Publisher. 2003; p. 145-184.
- [11] Mladenović N, Hansen P. Variable neighborhood search. *Computers & Operations Research*. 1997;24(11): 1097-1100.
- [12] Nikbakhsh E, Zegordi SH. A heuristic algorithm and a lower bound for the two-echelon location-routing

- problem with soft time window constraints. *Scientia Iranica Transaction E: Industrial Engineering*. 2010;17(1): 36-47.
- [13] Belenguer J, Benavent E, Prins C, Prodhon C, Calvo RW. A branch-and-cut method for the capacitated location-routing problem. *Computers & Operations Research*. 2011;38(6): 931-941.
- [14] Riera-Ledesma J, Salazar-González JJ. Solving school bus routing using the multiple vehicle traveling purchaser problem: A branch-and-cut approach. *Computers & Operations Research*. 2012;39(2): 391-404.
- [15] Ponboon S, Qureshi AG, Taniguchi E. Branch-and-price algorithm for the location-routing problem with time windows. *Transportation Research Part E: Logistics and Transportation Review*. 2016;86: 1-19.
- [16] Marinakis Y, Marinaki M, Migdalas A. A hybrid clonal selection algorithm for the location routing problem with stochastic demands. *Annals of Mathematics and Artificial Intelligence*. 2016;76(1-2): 121-142.
- [17] Golozari F, Jafari A, Amiri M. Application of a hybrid simulated annealing-mutation operator to solve fuzzy capacitated location-routing problem. *The International Journal of Advanced Manufacturing Technology*. 2013;67(5-8): 1791-1807.
- [18] Bay SD. Nearest neighbor classification from multiple feature subsets. *Intelligent Data Analysis*. 1999;3(3): 191-209.