



# A new geometric shape-based genetic clustering algorithm for the multi-depot vehicle routing problem

G. Nilay Yücenur, Nihan Çetin Demirel \*

*Yıldız Technical University, Mechanical Faculty, Industrial Engineering Department, Beşiktaş, Istanbul, Turkey*

## ARTICLE INFO

### Keywords:

Multi-depot vehicle routing problem  
Genetic algorithm  
Genetic clustering  
The nearest neighbor algorithm

## ABSTRACT

In this paper, a new type of geometric shape based genetic clustering algorithm is proposed. A genetic algorithm based on this clustering technique is developed for the solution process of the multi-depot vehicle routing problem. A set of problems obtained from the literature is used to compare the efficiency of the proposed algorithm with the nearest neighbor algorithm so as to solve the multi-depot vehicle routing problem. The experimental results show that the proposed algorithm provides a better clustering performance in terms of the distance of each customer to each depot in clusters. This result in a considerably less computation time required, when compared with the nearest neighbor algorithm.

© 2011 Elsevier Ltd. All rights reserved.

## 1. Introduction

Because it is found to be widely applicable to many real-world situations and is related to the logistics distribution problem, the vehicle routing problem (VRP) has been studied extensively. The VRP is a very important problem in research area because it is important in satisfying customer demands in an exact and timely manner (Demirel, Demirel, & Taşdelen, 2008). The VRP can be described as a problem of designing optimal delivery or collection routes from one or several depot(s) to a number of geographically scattered customers subject to side constraints, and the objective is to find a set of routes which minimizes the total distance traveled (Jeon, Leep, & Shim, 2007).

The VRP is categorized as HVRP (heterogeneous VRP), CVRP (capacitated VRP), VRPTW (VRP with time windows), SVRP (stochastic VRP), VRPB (VRP with backhauls), VRPPD (VRP with pickup and delivery) and so on with different constraints. Another variant of VRP is multi-depot vehicle routing problem (MDVRP), which is discussed in this paper.

The VRP and its variants are NP-hard problems. For the solution of the VRP, lots of exact algorithms, such as column generation, integer programming, branch and bound, branch and cut were described in Jin, Liu, and Eksioğlu (2008), Rousseau, Gendreau, and Feillet (2007), and Ralphs (2003). Because of their time consuming characteristics for some big size instances, the exact algorithms are losing their popularity. Thus, for large scale problem instances as typically found in industrial applications are solved with heuristics such as tabu search (Bolduc, Laporte, Renaud, & Boctor, 2010;

Brandao, 2009; Zachariadis, Tarantilis, & Kiranoudis, 2009), ant colony optimization (Fuellerer, Doerner, Hartl, & Iori, 2009; Yu, Yang, & Yao, 2009), particle swarm optimization (Ai & Kachitvichyanukul, 2009), genetic algorithm (Baker & Ayechew, 2003; Jeon et al., 2007) and simulated annealing (Tavakkoli-Moghaddam, Safaei, & Gholipour, 2006).

In MDVRP, there is more than one depot to serve customers, and this constitutes its main characteristic structure. MDVRP is one of the most encountered distribution problems in real-world situations, although it has not attracted so much attention as the other types of VRP. In MDVRP the customers have to be assigned to a certain depot; the routes can start at any depot but have to return to the same depot they started with.

For the solution of the MDVRP, two-phase solution techniques, such as cluster first-route second algorithm (Sweep algorithm, Fisher and Jaikumar algorithm, Bramel and Simchi-Levi, Petal algorithm, Taillard's algorithm) and route first-cluster second algorithm are used. In this paper, the first part of cluster first-route second algorithm is performed with genetic clustering. In this step the customers are assigned to the depot with the proposed new type of geometric shape based genetic clustering algorithm.

In the literature there are different genetic clustering approaches to solving different problems. Hanagangi and Nikolaou (1998) proposed a synergy between the cluster analysis technique, popular in classical stochastic global optimization, and the GA to accomplish global optimization; Liao, Ting, and Chang (2006) presented an adaptive genetic clustering method for exploratory mining of feature vector or time series; Laszlo and Mukherjee (2007) presented a genetic algorithm for selecting centers to seed the popular *k*-means method for clustering; Qing, Gang, Zaiyue, and Quiping (2008) also presented a genetic algorithm for selecting centers to seed the popular *k*-means method for clustering; and

\* Corresponding author. Tel.: +90 212 3832868.

E-mail addresses: [nserbest@yildiz.edu.tr](mailto:nserbest@yildiz.edu.tr) (G.N. Yücenur), [nihan@yildiz.edu.tr](mailto:nihan@yildiz.edu.tr) (N.Ç. Demirel).

Lin and Wei (2009) proposed a novel genetic algorithm based clustering approach for  $k$ -anonymization.

Although Salhi and Sari (1997) used an adaptive genetic clustering technique for the solution of the MDVRP, to our knowledge, no study concentrating mainly on the clustering step of the two-phase solution techniques for the MDVRP has been published in the genetic algorithm literature. This is the first study that proposes a new type of geometric shape based genetic clustering algorithm (with all genetic algorithm operators) for the clustering step of the vehicle routing problems with multi-depots.

The remaining parts of the paper are organized as follows: Section 2 gives brief information about genetic algorithm and discusses genetic clustering. Section 3 defines the application of a new geometric shape based genetic clustering algorithm to the MDVRP and presents the mathematical formulation of the problem. Moreover, in Section 3 a numerical example with different scenarios is illustrated, and solution techniques such as proposed geometric shape based genetic clustering and the nearest neighbor algorithms are performed. Computational results and analyses are discussed in Section 4; and Section 5 concludes the paper.

## 2. Genetic algorithm

Genetic algorithm (GA) is a stochastic optimization technique (Ho, Ho, Ji, & Lau, 2008; Samanlıoğlu, Ferrell, & Kurz, 2008). The basic idea of GA is to maintain a population of candidate solutions that evolves in a selection process. In GA terminology, the chromosome (string, individual) is the solution (coding), the genes (bits) are part of the solution, the locus is the position of the gene, alleles are the values of genes, the phenotype is the decoded solution and the genotype is the encoded solution (Gen & Cheng, 1997).

GA starts with an initial set of random solutions, called population. Each solution in the population is called a chromosome, which represents a point in the search space. The chromosomes evolve through successive iterations, called generations. During each generation, the chromosomes are evaluated using some measures of fitness. The fitter the chromosomes, the higher the probabilities of being selected to perform the genetic operations, including crossover and mutation. In the crossover phase, the GA attempts to exchange portions of two parents, that is, two chromosomes in the population to generate an offspring. The crossover operation speeds up the process to reach better solutions. In the mutation phase, the mutation operation maintains the diversity in the population to avoid being trapped in a local optimum. A new generation is formed by selecting some parents and some offspring according to their fitness values, and by rejecting others to keep the population size constant. After the predetermined number of generations is performed, the algorithm converges to the best chromosome, which hopefully represents the optimal solution or may be a near-optimal solution of the problem (Ho et al., 2008).

The attributes of GA is summarized as follows (Wang & Lu, 2009):

1. Genetic algorithm calculations are based on coded parameters, rather than the parameter values.
2. Genetic algorithms possess highly parallel search capability, avoiding becoming trapped in a local optimum.
3. Genetic algorithms have no complex mathematical formulas – only the fitness function must be calculated.
4. Genetic algorithms have no specific rules for guiding the search direction for an optimum; instead, a random search mechanism uses the probability rule.

In recent years, GA has been applied successfully to a wide variety of hard optimization problems, such as the traveling salesman

problem (Chang, Huang, & Ting, 2010; Liu & Zeng, 2009; Yang, Wu, Lee, & Liang, 2008), the vehicle routing problem (Cheng & Wang, 2008; Jeon et al., 2007), the quadratic assignment problem (Drezner, 2008; Misevicius, 2004) and scheduling problems (Gonçalves, Mendes, & Resende, 2008; Gu, Gu, Cao, & Gu, 2010; Shadrokh & Kianfar, 2007; Zhou, Cheung, & Leung, 2009).

### 2.1. Genetic clustering

Clustering is defined as a problem of classifying of patterns or data items into groups or clusters. A resulting partition should possess the following properties: (1) homogeneity within the clusters, i.e. data that belong to the same cluster should be as similar as possible and (2) heterogeneity between clusters, i.e. data that belong to different clusters should be as different as possible (Lin, Yang, & Kao, 2005). Application fields include statistics, mathematical programming (such as location selecting, network partitioning, routing, scheduling and assignment problems, etc.) and computer science (including, pattern recognition, learning theory, image processing and computer graphics, etc.) (Chiou & Lan, 2001).

In the literature there are lots of clustering algorithms. These methods can basically be classified into two categories: hierarchical and non-hierarchical. The hierarchical methods can be further divided into the agglomerative methods and the divisive methods. The agglomerative methods merge together the most similar clusters at each level and the merged clusters will remain in the same cluster at all higher levels. In the divisive methods, initially, the set of all objects is viewed as a cluster and at each level some clusters are binary divided into smaller clusters (Tseng & Yang, 2001). There are also many non-hierarchical methods. Well-known partitioning-based clustering methods include  $k$ -means (Laszlo & Mukherjee, 2007; Maulik & Bandyopadhyay, 2000),  $k$ -medoids (Liao et al., 2006), and fuzzy versions: fuzzy  $c$ -means (Göktepe, Altun, & Sezer, 2005; Mingoti & Lima, 2006; Tari, Baral, & Kim, 2009) and fuzzy  $k$ -means (Arima, Hakamada, Okamoto, & Hanai, 2008).

A genetic algorithm can be applied to the clustering problem in either of two fashions. One is to use genetic algorithm as a searching mechanism to fine-tune the setting of another clustering method. The other is to adapt genetic algorithm directly to the clustering problem. This way, genetic algorithm is the core of the clustering method, and the solution is encoded inside the chromosomes of genetic algorithm (Lin & Wei, 2009).

Generally in clustering problems, the number of clusters in a data set is not known, as in most real-life situations. None of these non-genetic algorithm based clustering algorithms is capable of efficiently and automatically forming natural groups from all the input patterns, especially when the number of clusters included in the data set tends to be large (Lin et al., 2005).

Under limited conditions, a genetic algorithm based clustering technique is expected to provide an optimal clustering structure with respect to the clustering conditions and variables that are considered.

## 3. Applying new geometric shape based genetic clustering algorithm to the MDVRP

The proposed algorithm is based on Thangiah and Salhi's (2001) adaptive clustering method. In the adaptive clustering method, the geometric shape which is mapped, namely, the circle, to the chromosome can be used effectively to route vehicles if the shapes have the capability to adapt to the route shapes, which result in the minimization of the routing cost. A new type of geometric shape based genetic clustering algorithm uses the circles. Differently from the adaptive clustering method, this algorithm is used only for the clustering step of the MDVRP solutions; it is not related

with the objective function. On the other hand, in this algorithm, differently from the adaptive clustering method, the circles are drawn from the depot with a random radius. The origin of the circle is the depot. Therefore, the number of circles is equal to the number of depots.

When circles are drawn from the depots in MDVRP problems, they do not overlap completely. There are three conditions that need to be considered when assigning customers to be clustered within a circle. A customer location can be inside a circle, it can be on the circumference of a circle, or it can be outside a circle. Fig. 1 represents an example obtained from two depots, A and B, and three customers,  $c_1$ ,  $c_2$ , and  $c_3$ . The placement process of the customers to the depots is the same as that of the adaptive clustering method. If a customer location is inside a circle, as  $c_1$  is in circle A, or on the circumference of a circle, as  $c_2$  is on circle A, it is assigned to that respective circle. If a customer is outside all the circles, then the distance of that customer location from each of the circumferences of the circles is calculated and the customer is assigned to the circle whose Euclidean distance is closest to it. In Fig. 1, the customer location  $c_3$  is outside the circles A and B. If the distance between  $c_3$  and the circumference of circle A is  $d_A$ , and between  $c_3$  and circle B is  $d_B$ , then  $c_3$  will be assigned to circle A if  $d_A \leq d_B$ , or to circle B if  $d_B < d_A$ .

In the proposed algorithm, genetic algorithm is used for the grouping of all the customers to the depots after the clustering process. In the clustering step, in order to draw circles with a random radius, the genetic algorithm optimization toolbox for MATLAB interface is integrated with the tools of package programs such as Visual Studio.NET 2008, .NET Framework 3.5, and C#. The user interface of the proposed new geometric shape based genetic clustering algorithm is represented in Fig. 2.

After the placement of the customers to the depots, the genetic algorithm process starts with all its operational steps. The customers are assigned to the depots by the circles with a random radius length. If  $m$  is the number of depots, there are  $2^m$  different lineup combinations which generate the population pool. For each population, the fitness value is calculated in order to obtain the objective function which minimizes the total traveled distance. After calculating the fitness value, all genetic algorithm operators are started to run respectively. Firstly, the chromosomes are selected, with the roulette wheel technique, from the binary-coded population, and running of the cyclical crossover operator is started. Secondly, by changing two neighbor genes, mutation operator is performed to offspring which are obtained from crossover. Following these two operators, the fitness value is calculated again. If the genetic algorithm process reaches the stop rule, which is the maximum number of produced generation, then the algorithm is stopped. Otherwise, all steps of the algorithm are repeated.

**Representation:** First, we encode decision variables into binary strings. In binary coding, each chromosome is defined with binary strings made up of 0 or 1. In Fig. 3, binary coding is represented.

**Initial population:** The initial population is generated with circles with a random-length radius.

**Selection:** The roulette wheel approach is used for the selection process. This approach belongs to the fitness-proportional selec-

tion and can select a new population with respect to the probability distribution based on fitness values. The roulette wheel approach can be constructed as follows (Gen & Cheng, 1997):

1. Calculate the fitness value  $eval(v_k)$  for each chromosome  $v_k$ :

$$eval(v_k) = f(x), \quad k = 1, 2, \dots, \text{population size}$$

2. Calculate the total fitness for the population:

$$F = \sum_{k=1}^{\text{pop.size}} eval(v_k)$$

3. Calculate selection probability  $p_k$  for each chromosome  $v_k$ :

$$p_k = \frac{eval(v_k)}{F}, \quad k = 1, 2, \dots, \text{population size},$$

4. Calculate cumulative probability  $q_k$  for each chromosome  $v_k$ :

$$q_k = \sum_{j=1}^k p_j, \quad k = 1, 2, \dots, \text{population size},$$

The selection process begins by spinning the roulette wheel population size times, and each time a single chromosome is selected for a new population.

**Crossover:** For a crossover operator, the cyclical crossover method is used. In this method, the first gene is selected from the first chromosome and located into the first offspring. From the second chromosome, the gene which is matched with the gene of the first chromosome is located into the second offspring. After these steps, all the genes are relocated in a cyclical shape. All the steps are repeated so as to obtain the second offspring. In Fig. 4, the cyclical crossover is represented.

**Mutation:** For a mutation operator, random method is used. In this method, two neighbor genes are randomly selected and then their locations are changed. In Fig. 5, the method of randomly changing two neighbor genes is shown.

### 3.1. The description of the problem and the mathematical model

This paper focuses on the multi-depot vehicle routing problem which involves more than one depot. Our objective is to minimize the total traveled distance between each depot and the customers which are assigned to the depot. The number and locations of the depots and customers are known. Each depot is large enough to store all the products which are ordered by the customers. Each vehicle route begins and ends at the same depot. Each customer is visited exactly once by a vehicle.

We need to cluster a set of customers to be served by the same depot, which brings us to the grouping problem, which is the subject of our paper. We group customers into the depots in 23 different problem sets which are obtained from the literature. In order to simplify the problem, we define the sets, parameters and variables used in the mathematical model as follows:

$N = (1, \dots, n)$  the set of customers;

$M = (1, \dots, m)$  the set of depots;

$C = (1, \dots, c)$  the set of circles;

$V = (1, \dots, v)$  the set of vehicles;

$P = N \cup M$  the set of all points;

$D_{ij}$  = Euclidean distance between customer  $i$  and  $j$  where  $i, j \in N$ ;

$d_i^l$  = Euclidean distance between customer  $i$  ( $i \in N$ ) and depot  $l$  ( $l \in M$ );

$L_c$  = Set of customers in circle  $c$  ( $c \in C$ );

$X_{ijk} = 1$ , if vehicle  $k$  travels directly from customer  $i$  to customer  $j$  ( $i, j \in N$ ); 0 otherwise.

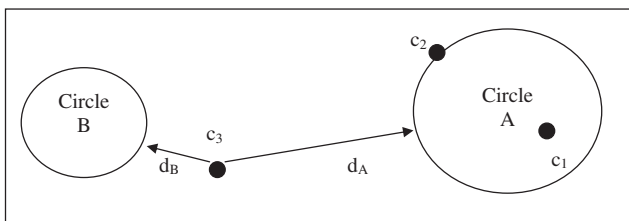


Fig. 1. The relationships between  $c_1$ ,  $c_2$ , and  $c_3$  with respect to circles A and B.

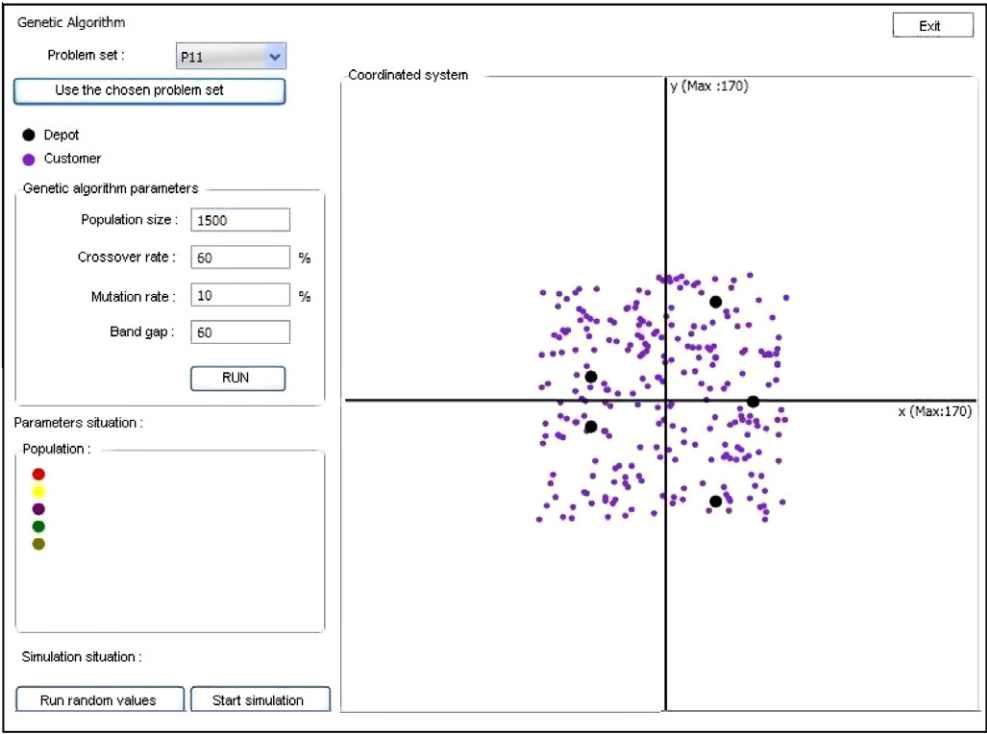


Fig. 2. The user interface for the proposed algorithm.

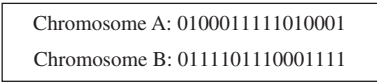


Fig. 3. Binary strings.

With this notation, the problem can be formulated as below:

$$\min Z = \sum_{i=0}^N \sum_{j=0, i \neq j}^N \sum_{k=1}^V \sum_{l=1}^M d_i^k x_{ijk}$$

subject to

$$\sum_{k \in V} \sum_{j \in P} x_{ijk} = 1 \quad (i \in N, j \in N, N \in P) \tag{1}$$

$$\sum_{i \in N} \sum_{j \in N} x_{ijk} \leq 1 \quad (k \in V) \tag{2}$$

$$\sum_{k \in V} \sum_{i=0, i \neq j, i, j \in N} x_{ijk} = 1, \tag{3}$$

$$\sum_{i \in N} x_{i0k} \leq 1 \quad (k \in V) \tag{4}$$

$$C = M \tag{5}$$

The objective function consists of the total distance between each depot and the customers which are assigned to the depot. Constraint (1) requires that each customer can be assigned to a sin-

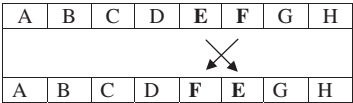


Fig. 5. Mutation with changing two neighbor genes randomly.

gle route. Constraint (2) ensures that one route can be served at most once. Constraint (3) requires that only one vehicle visits each customer only once. Constraint (4) requires that each vehicle starts from and finishes at the same depot, following one route only. Constraint (5) says that the number of circles is equal to the number of depots.

3.2. The definition of the problem

The proposed geometric shape based genetic clustering algorithm was applied to solve the 23 problems obtained from literature and used lots of researchers who studied about multi-depot vehicle routing problems (Cordeau, Gendreau, & Laporte, 1997; Fallahi, Prins, & Calvo, 2008; Renaud, Laporte, & Boctor, 1996; Thangiah & Salhi, 2001). In these problem sets the number of customers varies from 50 to 360, and the number of depots varies from 2 to 9 [1].

3.3. The nearest neighbor algorithm

In the nearest neighbor algorithm, the salesman starts his tour from some city and then visits the city nearest to the starting city.

Chromosome A:	9	8	2	1	7	4	5	10	6	3
Chromosome B:	1	2	3	4	5	6	7	8	9	10
Offspring 1:	9	2	3	1	5	4	7	8	6	10
Offspring 2:	1	8	2	4	7	6	5	10	9	3

Fig. 4. The cyclical crossover.

```

BEGIN
{
  compute  $A_{i,l}$ 
   $A_{i,l} = \sqrt{(x_i - x_l)^2 + (y_i - y_l)^2}$ 
   $i \in U$ ,  $i$  is not assigned to any depot,  $N \supset U$ 
  repeat
  {
    if
    {
       $\min \{d_i^l\}$  is calculated
    }
    then
      assign customer  $i$  to depot  $l$ 
    }
  } end if
} until
  When  $U = \emptyset$ 
}
END.

```

Fig. 6. The structure of the nearest neighbor algorithm.

From there he visits the nearest city that was not visited so far, until all cities are visited. At the end, the salesman returns to the start.

In this paper, in every problem set, all the distances between the depots and the customers are calculated and the customers are assigned to the nearest depot. The distance between two nodes is calculated with the Euclidean distance. The calculation of the distance between these two nodes can be formed depending on their coordinates.

$$A_{ij} = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2} \quad (6)$$

The structure of the nearest neighbor algorithm is represented in Fig. 6. In this definition,  $U$  is the set of customers which are not assigned to any depot.

### 3.4. Application for genetic clustering

The genetic clustering implementation used in the current study is based on the variant of genetic algorithm that differs from others with random-values-based circles. Furthermore, this is the first study that proposes a new type of geometric shape based genetic clustering algorithm (with all genetic algorithm operators) for the clustering step of the vehicle routing problems with multi-depots. The implementation steps are outlined in Fig. 7.

A new type of geometric shape based genetic clustering algorithm was formed with all genetic algorithm parameters. As a control operator, the population size, the crossover rate, the mutation rate and the band gap were used, and they were changeable by the user. In Fig. 8, there is a user interface of circles with random values, and the final solution of one example from the problem sets. The example solution belongs to Problem 11.

## 4. Computational results and analysis

This section represents the experimental results of 23 multi-depot vehicle routing problems. We compare our new type of geometric shape based genetic clustering algorithm results with the nearest neighbor algorithm and our proposed algorithm with different parameter values. The proposed algorithm was coded by C# on Intel® Core™ i7 Extreme Edition PC at 1333 MHz 500 GB 7200 RPM hard disk.

The evaluation is performed for the assigning step of the solution process of the multi-depot vehicle routing problems. First, the proposed algorithm results are compared with the nearest

```

BEGIN
{
  Initialize population by using random values for building circles  $P(0)$ 
  Set parameters by user
    Set parameter "population size"
    Set parameter "crossover rate"
    Set parameter "mutation rate"
    Set parameter "band gap"
  repeat
  {
    Evaluate chromosomes according to the fitness function
    Perform the selection operator to two chromosomes
      Perform "the roulette wheel selection operator"
    Perform the crossover operator
      Perform "the cyclical crossover operator"
    Perform the mutation operator
      Perform "the changing two neighbor genes mutation operator"
    Evaluate the offspring
    if
    {
      The offspring is better than the worst chromosome in population  $P$ 
    } then
      replace the worst chromosome with the offspring
    }
    } end if
    generation = generation + 1
  }
} until
  the process reaches the stop rule "the maximum number of produced generations"
Select the best chromosome as a result
}
END.

```

Fig. 7. The pseudo-code of the new type of geometric shape based genetic clustering algorithm.



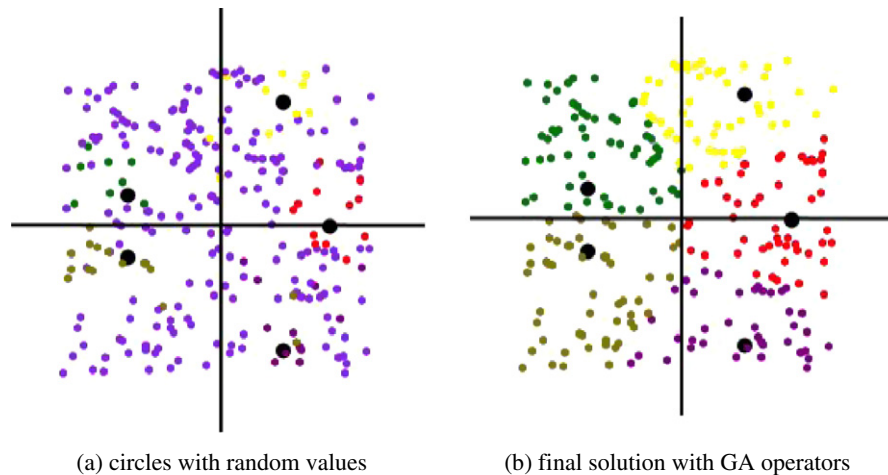


Fig. 8. The user interface for presenting the steps of the problem solution.

Table 1

A comparison of total distances with GA-clustering and the nearest neighbor algorithms.

Problem	Nearest neighbor algorithm	Geometric shape based genetic clustering CO rate 60%	Geometric shape based genetic clustering CO rate 3%	Error rate (%)
01	708	708	1637	2.31
02	708	708	1637	2.31
03	904	904	2185	2.42
04	1926	1926	4911	2.55
05	1949	1949	4307	2.21
06	1500	1500	4095	2.73
07	1445	1445	3901	2.69
08	14,472	14,472	33,286	2.30
09	11,739	11,739	31,108	2.64
10	10,434	10,434	29,319	2.81
11	9891	9891	23,837	2.40
12	2924	2924	7456	2.54
13	2924	2924	7456	2.54
14	2924	2924	7456	2.54
15	5794	5794	20,557	3.55
16	5794	5794	20,557	3.55
17	5794	5794	20,557	3.55
18	8691	8691	28,680	3.29
19	8691	8691	28,680	3.29
20	8691	8691	28,680	3.29
21	13,047	13,047	46,708	3.58
22	13,047	13,047	46,708	3.58
23	13,047	13,047	46,708	3.58

neighbor algorithm. In both approaches, the distance between the customers and the depots are calculated by Euclidean distance.

In the proposed algorithm, the circles are drawn from the depot with a random radius. The origin of the circle is the depot and the

number of circles is equal to the number of depots. After grouping the customers into the depots, the genetic algorithm is performed with all its operators. In the genetic algorithm process, as a control operator, the population size was set at 1500, the crossover rate was set at 60%, the mutation rate was set at 10% and the band gap was set at 60. The results obtained from the new type of geometric shape based genetic clustering algorithm are shown in Table 1, column 3.

The nearest neighbor algorithm was used in this paper for a comparison. This method is easy to implement and executes quickly, but it can sometimes miss shorter routes and classifications which are easily noticed with human insight. In the nearest neighbor algorithm, all distances between the depots and the customers are calculated and the customers are assigned to the nearest depot in every problem set. The results obtained from this algorithm are shown in Table 1, column 2.

As seen in Table 1, a new type of geometric shape based genetic clustering algorithm was able to obtain the best known solutions to 23 of the 23 problems, in comparison to the results obtained by the nearest neighbor algorithm. The results obtained from the new algorithm are equal to the results from the nearest neighbor algorithm.

In another comparison study, one of the control parameter values of the proposed algorithm was changed; in the new structure, the population size was set at 1500, the crossover rate was set at 3%, the mutation rate was set at 10% and the band gap was set at 60. The results obtained from the proposed algorithm with these parameter values are in Table 1, column 4. With these calculations, it is clear that the crossover rate is very important for the proposed algorithm structure and it has to be changed nearly 60% in order to obtain the correct results. After all these calculations, in Fig. 9, the

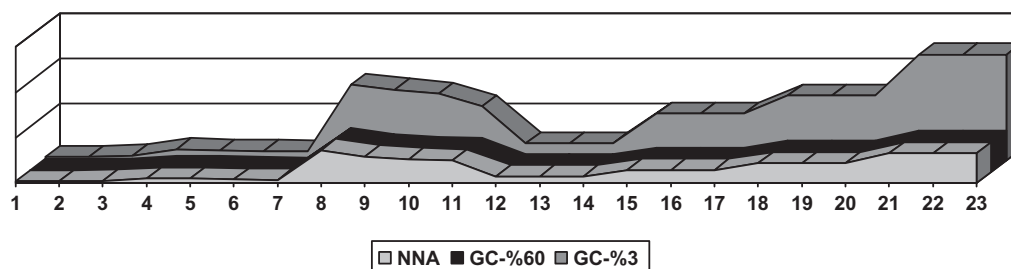


Fig. 9. The comparison chart of the results.

total distances obtained from different algorithms and parameter values are represented.

## 5. Conclusions

We proposed a new type of geometric shape based genetic clustering algorithm for the assigning step of the solution to a multi-depot vehicle routing problem. The proposed algorithm was tested on benchmark problems which were obtained from the literature, varying in size from 50 to 360 customers and 2–9 depots. To be able to compare the results, the problem sets were calculated according to the new proposed algorithm and the nearest neighbor algorithm. The algorithm proposed as a solution to the first step of the multi-depot vehicle routing problems, which is assigning the customers to depots, has proven to be equal to all the best solutions available, obtained from the nearest neighbor algorithm, and it does this by spending less computation time.

As a future work, we intend to extend our new genetic algorithm based assigning method with other meta-heuristics in order to solve each and every step of the multi-depot vehicle routing problems. Furthermore, the proposed geometric shape based genetic clustering algorithm can be extended with the other variants of the MDVRP problems.

## References

- Ai, T. J., & Kachitvichyanukul, V. (2009). A particle swarm optimization for the vehicle routing problem with simultaneous pickup and delivery. *Computers and Operations Research*, 36, 1693–1702.
- Arima, C., Hakamada, K., Okamoto, M., & Hanai, T. (2008). Modified fuzzy gap statistic for estimating preferable number of clusters in fuzzy *k*-means clustering. *Journal of Bioscience and Bioengineering*, 105(3), 273–281.
- Baker, B. M., & Ayechew, M. A. (2003). A genetic algorithm for the vehicle routing problem. *Computers and Operations Research*, 30, 787–800.
- Bolduc, M.-C., Laporte, G., Renaud, J., & Boctor, F. F. (2010). A tabu search heuristic for the split delivery vehicle routing problem with production and demand calendars. *European Journal of Operational Research*, 202, 122–130.
- Brandao, J. (2009). A deterministic tabu search algorithm for the fleet size and mix vehicle routing problem. *European Journal of Operational Research*, 195, 716–728.
- Chang, P.-C., Huang, W. H., & Ting, C. J. (2010). Dynamic diversity control in genetic algorithm for mining unsearched solution space in TSP problems. *Expert Systems with Applications*, 37, 1863–1878.
- Cheng, C.-B., & Wang, K.-P. (2008). Solving a vehicle routing problem with time windows by a decomposition technique and a genetic algorithm. *Expert Systems with Applications*, 36, 7758–7763.
- Chiou, Y.-C., & Lan, L. W. (2001). Theory and methodology genetic clustering algorithms. *European Journal of Operational Research*, 135, 413–427.
- Cordeau, J.-F., Gendreau, M., & Laporte, G. (1997). A tabu search heuristic for periodic and multi-depot vehicle routing problems. *Networks*, 30, 105–119.
- Demirel, T., Demirel, N. Ç., & Taşdelen, B. (2008). Time dependent vehicle routing problem with fuzzy traveling times under different traffic conditions. *Journal of Multiple Valued Logic and Soft Computing*, 14, 387–400.
- Drezner, Z. (2008). Extensive experiments with hybrid genetic algorithms for the solution of the quadratic assignment problem. *Computers and Operations Research*, 35, 717–736.
- Fallah, A. E., Prins, C., & Calvo, R. W. (2008). A memetic algorithm and a tabu search for the multi-compartment vehicle routing problem. *Computers and Operations Research*, 35, 1725–1741.
- Fuellerer, G., Doerner, K. F., Hartl, R. F., & Iori, M. (2009). Ant colony optimization for the two-dimensional loading vehicle routing problem. *Computers and Operations Research*, 36, 655–673.
- Gen, M., & Cheng, R. (1997). *Genetic Algorithms and Engineering Design*. A Wiley-Interscience Publication, John-Wiley & Sons, Inc.
- Gonçalves, J. F., Mendes, J. J. M., & Resende, M. G. C. (2008). A genetic algorithm for the resource constrained multi-project scheduling problem. *European Journal of Operational Research*, 189, 1171–1190.
- Göktepe, A. B., Altun, S., & Sezer, A. (2005). Soil clustering by fuzzy *c*-means algorithm. *Advances in Engineering Software*, 36, 691–698.
- Gu, J., Gu, M., Cao, C., & Gu, X. (2010). A novel competitive co-evolutionary quantum genetic algorithm for stochastic job shop scheduling problem. *Computers and Operations Research*, 37, 927–937.
- Hanagandi, V., & Nikolaou, M. (1998). A hybrid approach to global optimization using a clustering algorithm in a genetic search framework. *Computers Chemical Engineering*, 22(12), 1913–1925.
- Ho, W., Ho, G. T. S., Ji, P., & Lau, H. C. W. (2008). A hybrid genetic algorithm for the multi-depot vehicle routing problem. *Engineering Applications of Artificial Intelligence*, 21(4), 548–557.
- Jeon, G., Leep, H. O., & Shim, J. Y. (2007). A vehicle routing problem solved by using a hybrid genetic algorithm. *Computers and Industrial Engineering*, 53, 680–692.
- Jin, M., Liu, K., & Eksioğlu, B. (2008). A column generation approach for the split delivery vehicle routing problem. *Operations Research Letters*, 36, 265–270.
- Laszlo, M., & Mukherjee, S. (2007). A genetic algorithm that exchanges neighboring centers for *k*-means clustering. *Pattern Recognition Letters*, 28, 2359–2366.
- Liao, T. W., Ting, C.-F., & Chang, P.-C. (2006). An adaptive genetic clustering method for exploratory mining of feature vector and time series data. *International Journal of Production Research*, 44(14), 2731–2748.
- Lin, H.-J., Yang, F.-W., & Kao, Y. T. (2005). An efficient GA-based clustering technique. *Tamkang Journal of Science and Engineering*, 8(2), 113–122.
- Lin, J.-L., & Wei, M.-C. (2009). Genetic algorithm-based clustering approach for *k*-anonymization. *Expert Systems with Applications*, 36, 9784–9792.
- Liu, F., & Zeng, G. (2009). Study of genetic algorithm with reinforcement learning to solve the TSP. *Expert Systems with Applications*, 36, 6995–7001.
- Maulik, U., & Bandyopadhyay, S. (2000). Genetic algorithm-based clustering technique. *Pattern Recognition*, 33, 1455–1465.
- <http://neumann.hec.ca/chairedistributique/data/mdvrp/old/> (Arrival date: 16.12.2009).
- Mingoti, S. A., & Lima, J. O. (2006). Comparing SOM neural network with fuzzy *c*-means, *k*-means and traditional hierarchical clustering algorithms. *European Journal of Operational Research*, 174, 1742–1759.
- Misevicius, A. (2004). An improved hybrid genetic algorithm: New results for the quadratic assignment problem. *Knowledge-Based Systems*, 17, 65–73.
- Ralphs, T. K. (2003). Parallel branch and cut for capacitated vehicle routing. *Parallel Computing*, 29, 607–629.
- Renaud, J., Laporte, G., & Boctor, F. F. (1996). A tabu search heuristic for the multi-depot vehicle routing problem. *Computers and Operational Research*, 23, 229–235.
- Rousseau, L.-M., Gendreau, M., & Feillet, D. (2007). Interior point stabilization for column generation. *Operations Research Letters*, 35, 660–668.
- Qing, L., Gang, W., Zaiyue, Y., & Quiping, W. (2008). Crowding clustering genetic algorithm for multimodal function optimization. *Applied Soft Computing*, 8, 88–95.
- Salhi, S., & Sari, M. (1997). A multi-level composite heuristic for the multi-depot vehicle fleet mix problem. *European Journal of Operational Research*, 103, 95–112.
- Samanioğlu, F., Ferrell, W. G. Jr., & Kurz, M. E. (2008). A memetic random-key genetic algorithm for a symmetric multi-objective traveling salesman problem. *Computers and Industrial Engineering*, 55, 439–449.
- Shadrokh, S., & Kianfar, F. (2007). A genetic algorithm for resource investment project scheduling problem, tardiness permitted with penalty. *European Journal of Operational Research*, 181, 86–101.
- Tari, L., Baral, C., & Kim, S. (2009). Fuzzy *c*-means clustering with prior biological knowledge. *Journal of Biomedical Informatics*, 42, 74–81.
- Tavakkoli-Moghaddam, R., Safaei, N., & Gholipour, Y. (2006). A hybrid simulated annealing for capacitated vehicle routing problems with the independent route length. *Applied Mathematics and Computation*, 176, 445–454.
- Thangiah, S. R., & Salhi, S. (2001). Genetic clustering: An adaptive heuristic for the multi-depot vehicle routing problem. *Applied Artificial Intelligence*, 15, 361–383.
- Tseng, L. Y., & Yang, S. B. (2001). A genetic approach to the automatic clustering problem. *Pattern Recognition*, 34, 415–424.
- Wang, C.-H., & Lu, J.-Z. (2009). A hybrid genetic algorithm that optimizes capacitated vehicle routing problems. *Expert Systems with Applications*, 36, 2921–2936.
- Yang, J. Y., Wu, C., Lee, H. P., & Liang, Y. (2008). Solving traveling salesman problems using generalized chromosome genetic algorithm. *Progress in Natural Science*, 18, 887–892.
- Yu, B., Yang, Z.-Z., & Yao, B. (2009). An improved ant colony optimization for vehicle routing problem. *European Journal of Operational Research*, 196, 171–176.
- Zachariadis, E. E., Tarantilis, C. D., & Kiranoudis, C. T. (2009). A guided tabu search for the vehicle routing problem with two-dimensional loading constraints. *European Journal of Operational Research*, 195, 729–743.
- Zhou, H., Cheung, W., & Leung, L. C. (2009). Minimizing weighted tardiness of job-shop scheduling using a hybrid genetic algorithm. *European Journal of Operational Research*, 194, 637–649.