A Multi-objective Vehicle Routing Problem using Dominant Rank Method

Padmabati Chand
School Of Computer Engineering
KIIT University, Bhubaneswar, India

J. R. Mohanty
School of Computer Application
KIIT University, Bhubaneswar, India

ABSTRACT
Vehicle Routing Problem (VRP) is a NP-Complete and a multi-objective problem. The problem involves optimizing a fleet of vehicles that are to serve a number of customers from a central depot. Each vehicle has limited capacity and each customer has a certain demand. Genetic Algorithm (GA) maintains a population of solutions by means of a crossover and mutation operators. We propose new methods for genetic operators. The proposed method for crossover is Sub Route Mapped Crossover Method (SMCM) and for mutation is Sub Route Exchange Mutation Method (SEMM). This paper applies Dominant Rank method to get Pareto Optimal Set. The vehicle routing problem is solved with two objectives i.e. number of vehicles and total cost (distance). The proposed Dominant Rank Method finds optimum solutions effectively.

General Terms
Optimization, Customer, Vehicle.

Keywords
Vehicle Routing Problem, genetic algorithm, multi-objective optimization, dominant rank method, sub route mapped cross over method (SMCM), sub route exchange mutation method (SEMM).

1. INTRODUCTION
The vehicle routing problem (VRP) is a well-known NP-hard optimization problem which is encountered frequently in distribution logistics and transportation systems. It is a Multi Constrained Optimization Problem [2, 7]. VRP formulations are used in distribution services like post, parcel, and etc. Fisher [1] describes the problem as the efficient use of a fleet of vehicles, which must make a number of stops to pick up and deliver passengers or products. The term customer is used to denote the stops to pick up and deliver the product. Every customer is assigned exactly one vehicle in a specific order, which is done with respect to the capacity in order to minimize the total cost [3]. This paper studies the VRP as a multi-objective optimization problem (MOP), as implemented within a genetic algorithm (GA). The VRP is optimized in two objectives: number of vehicles and total distance. So the result of MOP technique is not only one solution, it is a set of solutions. This set is called Pareto Optimal Set, which consists of all the non-dominated solution. Dominant rank method is used to get Pareto Optimal Set. To improve the computational efficiency of GA, an improved mutation (Sub Route Exchange Mutation Method (SEMM)) and cross over (Sub Route Mapped Crossover Method (SMCM)) operation is exploited so that more parents' excellent performance can be inherited by off-springs.

The rest of the paper is organized as follows - section 2 gives a back ground study of genetic algorithm, multi objective optimization and need of genetic algorithm to solve VRP, section 3 gives problem formulation of VRP, chromosome representation, fitness evaluation and genetic operators (SMCM and SEMM), tournament selection and experimental results.

2. BACKGROUND DETAILS

2.1 Genetic Algorithm
In the GA, each chromosome in the population pool is transformed into number of routes [3, 4]. The chromosomes are then subjected to an iterative evolutionary process until a minimum possible number of route clusters is attained or the termination condition is met [5]. The evolutionary process is carried out as in ordinary GA using genetic operators (crossover, mutation) and selection operations on chromosomes for reproduction. The primary objective of the reproduction operator is to make duplicates of good solutions and eliminate bias solutions in a population, while keeping the population size constant. This is achieved by performing the following tasks:

1. Identify good (usually above average) solutions in a population.
2. Make multiple copies of good solutions.
3. Eliminate bad solutions from the population so that multiple copies of good solutions can be placed in the population.

At every generation stage, we need to select parents for mating and reproduction. A problem specific crossover operator that ensures solutions generated through genetic evolution is proposed. Hence both checking of the constraints and repair mechanism can be avoided, thus resulting in increased efficiency. A cross over operator is applied next to the strings of the mating pool. A little thought will indicate that the reproduction operator cannot create any new solutions in the population. It only makes more copies of good solutions at the expense of not-so-good solutions. The creation of new solutions is performed by crossover and mutation operators. Like the reproduction operator, there exists a number of crossover operators in the GA literature, but in almost all crossover operators, two strings are picked from the mating pool at random and some portion of the strings are exchanged between the strings to create two new strings. The crossover operator is mainly responsible for the search aspect of genetic algorithms, even though the mutation operator is also used for this purpose. The bit wise mutation operator changes a 1 to 0 and vice versa, with a mutation probability. The need for mutation is to keep diversity in the population. Working of GA is shown in the following algorithm.

Genetic Algorithm

Start

Step 1: Randomly initialize the population

Step 2: Set maximum generation

Step 3: Initialize generation as one

Step 4: While (generation <=maximum generation)

{

Step 4(a): Evaluate fitness function of each solution in the population

Step 4(b): Apply selection procedure to select set of solutions from population ,for next generation

Step 4(c): Apply genetic operators (crossover and mutation)

Step 4(d): generation =generation+1

}

End

2.2 Multi Objective Optimization

Multi-objective optimization, also known as multi-criteria optimization, is the process of simultaneously optimizing two or more conflicting objectives subject to certain constraints. If a multi-objective problem is well formed, there should not be a single solution that simultaneously minimizes each objective to its fullest. In each case we are looking for a solution for which each objective has been optimized to the extent that, if we try to optimize it any further, then the other objective(s) will suffer as a result [6]. Finding such a solution, and quantifying how much better this solution (compared to other such solutions) is the goal when setting up and solving a multi-objective optimization problem [3].When solving multi-objective problems our main focus is to calculate fitness value of individual solution and then selection means set of solutions for genetic operators. These set of solutions is called Pareto Optimal Set and fitness of solution is called Pareto approximation. It is important to find as many Pareto-optimal solutions as possible in a problem. Thus there are two goals in a multi-objective optimization. First goal is to find a set of solution as close as possible to the Pareto-optimal front and the second goal is to find a set of solutions as diverse as possible.

2.3 Need of Genetic Algorithm to Solve VRP

Every good optimization method needs to balance the extent or exploration on information obtained up until the current generation through recombination and mutation operators with the extent of exploitation through the selection operator. If the solutions obtained are exploited too much, premature convergence is expected. On the other hand, if too much stress is given on a search, the information obtained thus far has not been used properly. Therefore the solution time may be enormous and the search exhibits a similar behavior to that of a random search. Most classical methods have fixed transition rules and hence have fixed degrees of exploration and exploitation. Since these issues can be controlled in a GA by varying the parameters involved in the genetic operators, GA provide an ideal platform for performing a flexible search in VRP. Genetic algorithm is widely used to solve optimization problems for its characteristic, especially vehicle routing problem [11, 12].

3. PROBLEM FORMULATION

3.1 Description of VRP

Vehicle Routing Problem (VRP) has been considered as a significant segment in logistic handling. Thus, a proper selection of vehicle routes plays a very important part to ameliorate the economic benefits of logistic operations [10]. It is used to design least cost routes from a central depot to a set of geographically dispersed points (customers, stores, schools, cities, warehouses, etc.) with various demands [7, 8]. We present the problem description of the Vehicle Routing Problem (VRP) as follows.

- A set C = $\{c_1, c_2, \ldots, c_n\}$ of customers, with known demands $d_i > 0$,where i=2,3,…n
- A special node c_0, called the depot.
- d_{ij} is the distance of the arc(i,j). The arc start and end in the same depot.
- A fleet of K(k=1,k=2…k=m) identical vehicles available in the depot.
- Customer should be visited in such a way that vehicle capacity constraint (v_c) should not violated.
- Each customer should visit once by one vehicle.

We have to design routes, in such a way that there should be minimum number of distance and minimum number of vehicle. Two fitness functions are taken here - first fitness function (F1) to minimize distance and the second fitness function (F2) to minimize number of vehicles.

F = minimum(F1,F2)

F1 = minimum(distance)

$$= \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} d_{ij}$$

F2=minimum(vehicle)

$$= \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \sum_{k=1}^{m} v_{ij}^{k}$$

Where

i and j are customers in the population.

n is the total number of customers in the population(or population size).

m is the total number of vehicles available in the depot.

$v_{ij}k$ indicates number of vehicles (k number of vehicles) required to traverse the arc (i,j).

The following figure shows a simple graphical model of the VRP and its solution. In this figure, fleet of identical vehicles (number of vehicle four, with each vehicle capacity is 50) available in the depot (d). Number of customers, numbered from 1 to 10, with demand of each customer is illustrated in the figure. In “Table 1” distance from depot to customer, customer to depot, customer to customer explained. By considering constraints and objective (minimum distance and minimum number of vehicles), the exact routes explained in “Table 2”. Distance of each sub routes, total route distance and total number of vehicles required to traverse each customer illustrated in “Table 2”. The final route design is explained in “Figure 2”.

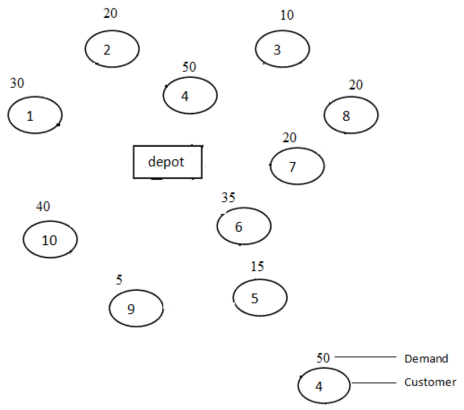


Fig 1: Customers with Demand

Table 1. Distance table

	Depot	1	2	3	4	5	6	7	8	9	10
Depot	-	5	6	10	3	15	4	5	10	15	5
1	5	-	2	10	5	20	15	25	30	30	5
2	6	2	-	5	2	20	10	20	20	30	20
3	10	10	5	-	4	20	15	5	4	30	30
4	3	5	2	4	-	30	20	15	10	30	20
5	15	20	20	20	30	-	5	10	20	5	10
6	4	15	10	15	20	5	-	2	5	10	20
7	5	25	20	5	15	10	2	-	2	20	30
8	10	30	20	4	10	20	5	2	-	30	40
9	15	30	30	30	30	5	10	20	30	-	5
10	5	5	20	30	20	10	20	30	40	5	-

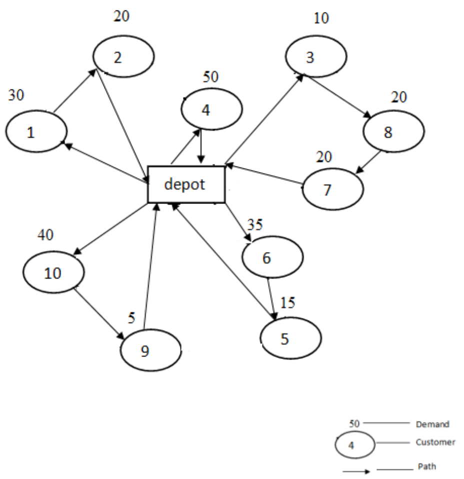


Fig 2: Final Route Design

Table 2. Sub routes with distance and number of vehicles

Sub Routes	Distance	Vehicle
d-1-2-d	13	1
d-4-d	6	1
d-3-8-7-d	21	1
d-6-5-d	24	1
d-9-10-d	25	1
Total	**89**	**5**

3.2 Chromosome Representation and Initial Population Creation

In our approach, a chromosome has two parts. First part of the chromosome represents routes. In "Figure 3" total number of route is 3. A gene in a given chromosome indicates the original node number assigned to a customer, while the sequence of genes in the chromosome indicates the order of visitation of customers. Thus, the chromosome consists of integers, where new customers are directly represented on a chromosome with their corresponding index number and each committed customer is indirectly represented within one of the groups [9]. The second part of the chromosomes indicates last node number of each sub route. In the following "Figure 3", sub routes are there and the second part of the chromosome (1,7,4) are last node numbers of three sub routes.

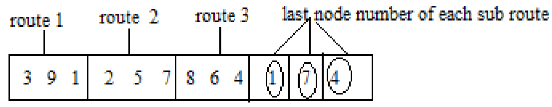


Fig 3: Chromosome Representation

3.3 Fitness Evaluation

When dealing with multi-objective problem, fitness assignment cannot be done directly, as the number of objective function is more than one. In this paper we consider dominance approaches to assign fitness of individual solution and we named this method as dominant rank method. In the following we have explained two algorithms. First algorithm is dominant rank method and the second is dominant rank method with GA.

3.3.1 Dominant Rank Algorithm

Start

Step 1: Initialize population

Step 2: Repeat generation until to reach maximum generation

{

Step 2(a): Calculate distance and number of vehicles of each solution in the population

Step 2(b): Count number of solutions dominated by certain individual. If j number of solutions dominated by solution i then fitness of i is j

Step 2(c): Repeat step 2(a) and 2(b) to get dominance relationship of rest of the solutions in the population

}

End

3.3.2 Dominant Rank Method with GA Algorithm

Start

Step 1: Read problem instance data

Step 2: Set GA parameters

Step 3: Generate randomly an initial population

Step 4: For Generation =1 to Maximum Generation

{

Step 4(a): Evaluate fitness of the individuals of population

Step 4(b): Apply Dominant rank method and select new population (Tournament Selection)

Step 4(c): Apply GA operators (crossover (SMCM) and mutation (SEMM))

Step 4(d): generation =generation+1

}

End

3.3.3 *Diversity Method*

In Pareto approximation, diversity is important because all the solutions are different. Density information gives us good metric to increase this diversity. This means probability to select solution decreases the greater density of solutions in its neighborhood. So for density information histogram "Figure 4" method is applied. Histogram is the technique consists of equal dimension grids. According to fitness value different solutions are located in different grids. Maximum number of solutions in any particular grid is the density information.

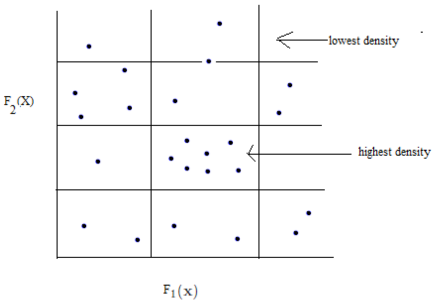


Fig 4: Histogram Method

3.4 Cross Over (Sub Route Mapped Crossover Method (SMCM))

Initial experiments using standard crossover operators such as Partially-Mapped-Crossover (PMX) and uniform order crossover (UOC) yielded non-competitive solutions [10]. Hence, a problem-specific crossover operator is utilized that generates feasible route schedules [9]. Our proposed crossover technique explained in "Figure 5". According to the figure, two parents A and B are selected from the population. We have considered two chromosomes and in each chromosome there are 9 customers. Number of sub routes in chromosome 1 is 3 and in chromosome 2 is 4.

Chromosome 1: 3 9 1- 2 5 7-8 6 4

Chromosome 2: 1 5 2-9 3 8-6-4 7

The last node number of each sub route of chromosome 1 is the first sub route in offspring 1 i.e, 1 7 4, by satisfying all the constraints. If the constraints are not satisfied then make a new route. Similarly the last node number of each sub route of chromosome 2 is the first sub route of offspring 2 i.e. 2,8,6,7 by satisfying all the constraints. Since constraints are not satisfied so here one sub route is 2 8 and another is 6 7.

Offspring 1: 1 7 4

Offspring 2: 2 8 -6 7

Delete the visited node number which is in offspring 1 from chromosome 2. Similarly delete the visited node number which is in offspring 2 from chromosome 1. After deletion rest of the nodes in sub routes are:

Chromosome 1: 3 9 1-5 -4

Chromosome 2: 5 2-9 3 8-6

Now second sub route of offspring 1 is, last node number of each sub route in chromosome 2, by satisfying all the constraints. If the constraints are not satisfied then make a new route. Same procedure follows for offspring 2.

Offspring 1: 1 7 4 – 2 8 6

Offspring 2: 2 8 -6 7- 1 5 4

Delete the visited node number which is second sub route of offspring 1 from chromosome 2. Similarly delete the visited node number which is second sub route of offspring 2 from chromosome 1. So after deletion rest of the nodes in sub routes are:

Chromosome 1: 3 9

Chromosome 2: 5 - 9 3

Insert sequentially rest of the node number in offspring 1 from chromosome 2. If constraints are not satisfied then make new sub routes. Same method follows for offspring 2. So the final routes are:

Offspring 1: 1 7 4 – 2 8 6 – 5 9 3

Offspring 2: 2 8 -6 7 3- 1 5 4 – 9

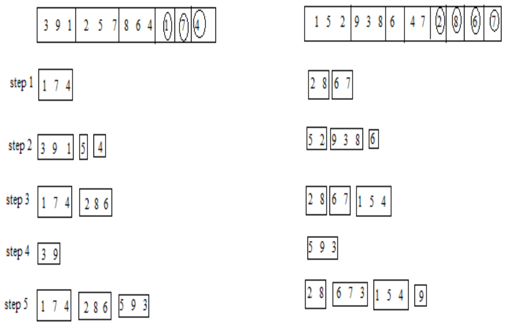
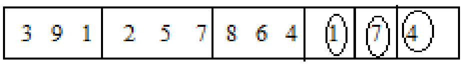


Fig 5: Sub Route Mapped Crossover Method

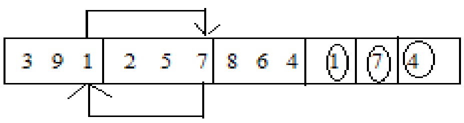
3.5 Mutation (Sub Route Exchange Mutation Method (SEMM))

In this method two sub routes selected randomly, then last node number of each sub routes exchanged, by satisfying all the conditions [2]. The following figure explain mutation step wise.

Step 1: The following chromosome are taken for mutation where 1 7 4 are last node numbers of each sub routes.



Step 2: Suppose randomly two sub routes are selected i.e. 2 5 7 and 3 9 1. Then exchange last node number of each sub routes.



Step 3: After exchanging, the new chromosome is:

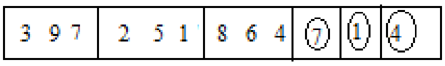


Fig 6: Sub Route Exchange Mutation Method

3.6 Tournament Selection

At every generation stage, we need to select parents for mating and reproduction [2]. Tournament selection is used to perform fitness-based selection of individuals for further evolutionary reproduction [4]. In the tournament selection, tournaments are played between two solutions and the better solutions is chosen and placed in the mating pool. Two other solutions are picked again and another slot in the mating pool is filled with the better solution. If carried out systematically, each solution can be made to participate in exactly two tournaments. The best solution in a population will win both times, thereby making two copies of it in the new population. Using similar argument, the worst solution will lose in both tournaments and will be eliminated from the population. In this way any solution in a population will have zero, one or two copies in the new population. It has been shown by Goldberg and Deb in literature that the tournament selection has better or equivalent convergence and computational time complexity properties when compared to any other reproduction operator [5]. Tournament selection operator is that just by changing the comparison operator, the minimization and maximization problems can be handled easily and VRP is a minimize optimization problem.

3.7 Experimental Results

This section describes computational experiments carried out to investigate the performance of the proposed GA. By our given fitness function, it minimizes both number of vehicles and travel costs without bias. All the programs are implemented in mat lab. The program was run on an Intel Pentium IV 1.6 MHz PC with 512 MB memory. We have applied histogram method for selection of number of solutions. Then we applied tournament selection, where tournament size is two and genetic operators like for crossover SMCM and for mutation SEMM. The probability of cross over is 0.8 and probability of mutation rate is 0.2. In this program we used Solomon's benchmark data set. The following figures illustrate the progress of the genetic algorithm. "Table 3" represents experimental results. In "Table 3" we are taken different solomon's data set. Basically the keywords in the data set represent how the customers are located. c represents customers located in clustered manner, r means customers are in random wise, rc means customers are in random and clustered manner. So c101, c105 and rc101 represents hundred different customers are located in clustered and random clustered manner. c201, r201, rc201 represents two hundred different customers located in clustered, random and random clustered manner. "Figure 7" represents initial population of the instance c105, "Figure 8" represents network topology for the instance c105 and Pareto optimal front explained for the instance c105 in "Figure 9".

Table 3. Experimental results

Solomon data set	Vehicle	Distance
c101	13	1467.43
c105	9	1456.23
c201	3	773.21
r201	6	1724.43
rc101	15	1987.12
rc201	4	2453.10

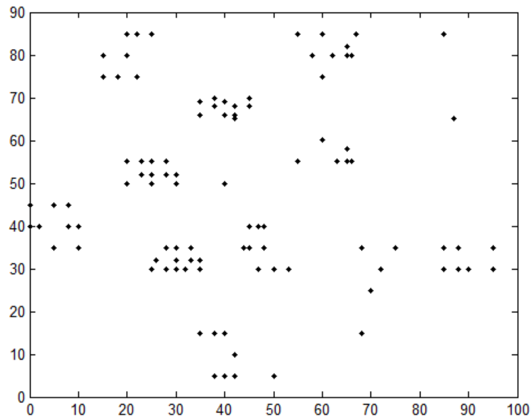


Fig 7: Initial Population for Instance c105

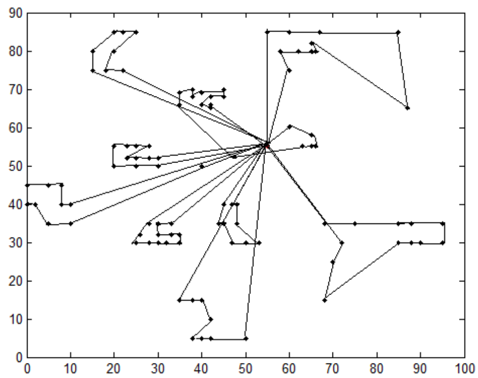


Fig 8: Result for Instance c105

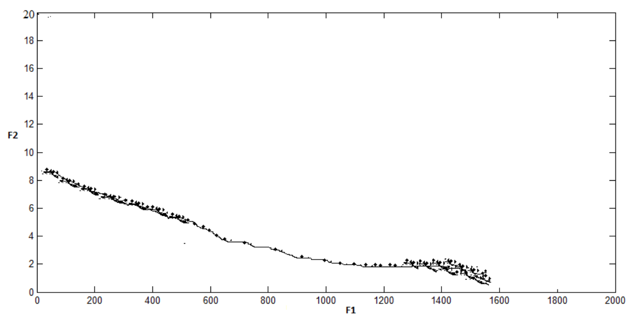


Fig 9: Pareto Optimal Front for Instance c105

4. CONCLUSION

In this paper we have presented a GA based approach for the VRP. The approach was tested using problem instances reported in the literature, derived from publicly available Solomon's benchmark data for VRP. To get Pareto Optimal Set, Dominant rank method is used. Here two new methods are proposed for genetic operators - One method for cross over SMCM and another for mutation SEMM. The experimental results revealed that GA is able to determine the optimum route for the vehicles while maintaining their constraints of capacity. Our future work is the comparison of VRP with GA and other classical optimization methods.

5. REFERENCES

[1] Fisher, M.1995 Vehicle routing. Handbooks of Operations Research and Management Science, vol. 8. Elsevier , pp. 1-33.

[2] Ombuki, B., Ross, J., Hanshar, F.2006. Multi-Objective Genetic Algorithms for Vehicle Routing Problem with Time Windows. Journal of Applied Intelligence, Springer Science, Vol. 24, pp.17-30.

[3] Haupt, R.L., Haupt, H.S. 2004 Practical Genetic Algorithms. 2nd Edition, A John Wiley & Sons, Inc., Publication.

[4] Deb, K.2001 Multi- Objective Optimization Using Evolutionary Algorithm. John Wiley & Sons, Ltd.

[5] Goldberg, D.E., Deb, K. 1991 A comparative analysis of selection schemes used in genetic algorithm. In G.J.E. Raslins(Ed.), Foundations of genetic algorithms, pp.69-93.

[6] Goldberg, D.E. 1998 Genetic Algorithm in Search, Optimization, and Machine Learning. Addison-Wesley, Reading, Massachusetts.

[7] Tan. K. C., Chew. Y.H. 2006. A Hybrid Multi objective Evolutionary Algorithm for Solving Vehicle Routing Problem with Time Windows. Computational Optimization and Applications, Springer Science, Vol. 34, pp.115–151.

[8] Chand, P., Mohanty, J.R. 2011. Multi Objective Genetic Approach for Solving Vehicle Routing Problem with Time Window. In proceeding of trends in Computer Science, Engineering and Information Technology, LNCCIS (CCSEIT), Vol.204, pp.336-343.

[9] Chand, P., Mohanty, J.R. 2011. Multi Objective Genetic Approach for Solving Vehicle Routing Problem. In Proceeding 4th International Conference on Computer Science and Information Technology (ICCSIT), in press.

[10] Mohammed, M.A., Ahmad, M.S., Mostafa, S.A. 2012. Using Genetic Algorithm in implementing Capacitated Vehicle Routing Problem. In Proceeding of International Conference On Computer & Information Science(ICCIS),Vol.1, pp.257-262.

[11] Zongyan, X., Haihua, L.,Wang, Y. 2011. An Improved Genetic Algorithm for Vehicle Routing Problem. In Proceeding of International Conference on Computational and Information Sciences (ICCIS), pp.1132-1125.

[12] Jiajiao, Q.,Yong, Z., Jianlin, M., Lixia, F. 2011. A new coding method for genetic algorithm in vehicle routing problem. In Proceeding International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), pp.201-204.