◆ Research Papers

A 3-D Genetic Algorithm for Finding the Number of Vehicles in VRPTW

Paik, Si Hyun *
Ko, Young Min *
Kim, Nae Heon **

Abstract

The problem to be studied here is the minimization of the total travel distance and the number of vehicles used for delivering goods to customers. Vehicle routes must also satisfy a variety of constraints such as fixed vehicle capacity, allowed operating time. Genetic algorithm to solve the VRPTW with heterogeneous fleet is presented.

The chromosome of the proposed GA in this study has the 3-dimension. We propose GA that has the cubic-chromosome for VRPTW with heterogeneous fleet. The newly suggested 'Cubic-GA (or 3-D GA)' in this paper means the 2-D GA with GLS(Genetic Local Search) algorithms and is quite flexible. To evaluate the performance of the algorithm, we apply it to the Solomon's VRPTW instances. It produces a set of good routes and the reasonable number of vehicles.

Keyword : VRP(Vehicle Routing Problem), VRPTW(Vehicle Routing Problem with Time Windows), VRPTWHtF(Vehicle Routing Problem with Time Windows and Heterogeneous Fleet), GA (Genetic Algorithm), GLS(Genetic Local Search)

1. Introduction

Vehicle routing problems comprise an interesting and important class of combinatorial problems. Their economic importance is marked by their presence in many areas of the manufacturing and logistics. The problem to be studied here is the minimization of the distance and the vehicles used for delivering a set of goods into each customer with vehicles. That is called VRP. The classical VRP involves a set of delivery cities to be serviced by a set of vehicles at a distribution center. There are many variations of the problem. The basic components of the problem are a fleet of vehicles with fixed capabilities (time, capacity, distance, etc.). The objective of the VRP is to develop a set of routes such that all delivery cities are serviced, the demands of the points assigned to each route do not violate the capacity of the vehicle which services the route, and the total distance by all vehicles is minimized.

VRPTW is more complex as it involves servicing customers with time windows using vehicles.

---

* Ph.D. student, Industrial Engineering department of Ajou University

** Professor, Industrial Engineering department of Ajou University

So each vehicle must visit a city at interval time[earliest time, latest time]. The constraints of the problem require the vehicles to service the customers while not overloading the vehicles, and to visit the customers after the earliest service time and before the latest service time. A vehicle has to wait, if one arrives at a customer location before the earliest service time and has to wait until the customer is ready for service. On the contrary a vehicle that arrives at a customer after the latest service time is considered to be tardy. The objective of the VRPTW is to obtain feasible routes while minimizing the number of vehicles and the distance traveled by the vehicles. In this paper GA(Genetic Algorithm) to solve the VRPTW(Vehicle Routing Problems with Time Window) with heterogeneous fleet is presented.

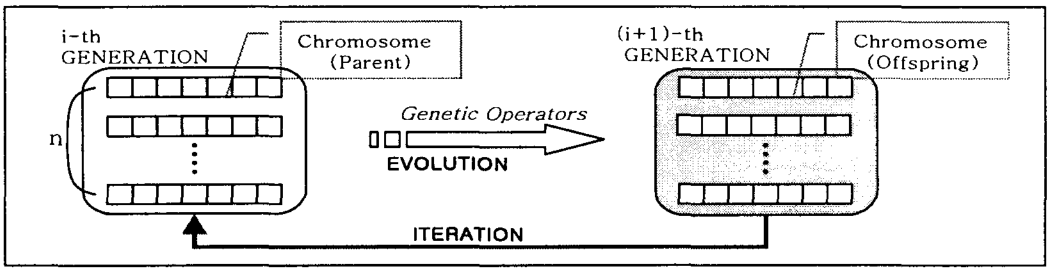
Recently, GAs have been widely reckoned as a useful vehicle for obtaining high quality or even optimal solutions for a broad range of combinatorial optimization problems. Unlike classical GA, the chromosome of the proposed GA has the new structure that is 3-dimensional. The suggested 'Cubic-GA' in this paper means the 2-D GA including GLS algorithms[5, 16].

A few researchers have studied VRPTW using GA. But they assumed homogeneous fleet. There is no work for VRPTWHtF using GA. We proposed GA which has the cubic-chromosome for VRPTW with heterogeneous vehicles.

The paper is organized as follows. Section 2 describes the several works in VRP and GA. In section 3, the cubic-GA used to solve the VRPTW with heterogeneous fleet is presented. Results for experiment are presented in Section 4. Section 5 concludes this paper and outlines areas for future research.

2. Previous Works

It is called the routing problem in logistic system 'Hub-and-Spoke system'[1]. Routing problems can be partitioned into several problems, i.e. VRP, QAP(Quadratic Assignment Problem), TSP(Traveling Salesman Problem), etc. We have focused on VRP in this paper. Since the VRPTW is NP-hard, most of researches have focused on heuristic approaches or case studies. Though optimal solutions to VRPTW can be obtained using exact methods, the computational time required to solve a VRPTW to optimality is huge. There are many literatures on vehicle routing problem with time window constraints. The classification of the exact approaches for VRPTW suggested by Fisher, Jrnsten and Madsen[4] is given below.



<Figure 1> Schematic diagram of GA

① Approaches based on dynamic programming.

② Approaches based on column generation

③ Lagrangian decomposition based methods

④ K-tree based methods

GAs have been deeply investigated in the last decade as a possible method for solving function optimization and combinatorial problems. The idea of GA was to evolve a population of candidate solutions to a given problem, using genetic operators, crossover and mutation, inspired by natural genetic variation and natural selection at each generation. A schematic diagram of the basic structure of a genetic algorithm is shown in figure 1.

Many researchers in GAs have worked how to represent a special problem, to decide the value of parameters (the crossover rate, the mutation rate, and population size), to redesign the genetic operators for the purpose of a special problem, and to develop new fit genetic operators.

In a view of representation, Prinetto, Rebaudengo and Reorda[19] have summarized how to represent a tour in GAs. They distinguished the methods in three ways below.

Path Representation; Ordinal Representation Adjacency Representation

Thangiah[20] studied the representation of the attributes (the origin and the radius of the circle) of a circle in a chromosome for clustering. The chromosome of GA was encoded with two-dimension in several works[10, 12, 16, 23].

In a view of crossover operators, there are OX(Order Crossover)[2], CX(Cycle Crossover)[18], HC(Heuristic Cross over)[19], MX(Matrix Crossover)[9], ER(Edge Recombina-tion operator)[8, 20, 22], CSEX(Complete Subtour Exchange Cross over)[11], SXX(Subsequence eXchange Crossover)[23], MPX (Maximal Preservative Crossover)[15], PMX(Partially Matched Crossover)[6,7], DPX(Distance Preserving Crossover) [14],THX(Time Horizon exchange crossover)[12].

Since the chromosome adapted in this paper has cubical structure, and includes more information.

3. Cubic-Genetic Algorithm

We mapped the X-axis to the capacity of vehicle, the Y-axis to the number of vehicles and the Z-axis to time window for VRPTW. The Cubic-GA in this paper means the 2-dimensional(2-D) GA with GLS algorithm[5, 16]. The logic of Cubic-GA is similar to usual GA except containing more information. The Cubic-GA defined with the 2-D GA and GLS(Genetic Local Search)[5, 16] is given below.

1)Population

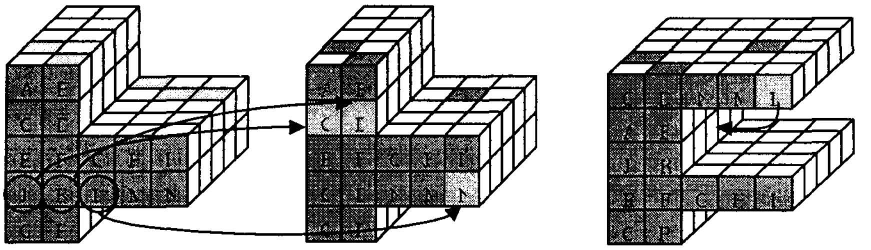
Population, with size n, is randomly generated

2) Selection

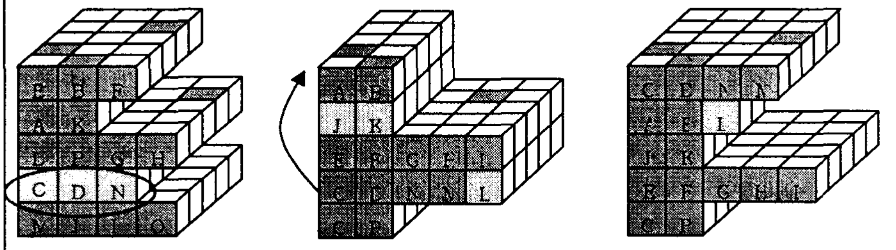
Two chromosomes are randomly selected using elitism for crossover.

3) Crossover (Considering X-Y axis (figure 2))

Revised PMX(Partially Matched Crossover) operator. is used for generating offspring(figure 2).



(a) Parent 1 (c) Offspring with redundant gene (e) Exchanging the order of vehicles



(b) Parent 2 (d) Offspring with replacement (f) Offspring considering capacity

<Figure 2> Revised PMX operator

Select one chromosome from 2 parent chromo- somes(Fig. 2 (a), (b)).

- We assume that the parent 2 is selected between parent 1 and parent 2.

Select a row(building block) randomly among of the rows of the parent 2.

- The mark of circle in figure 2(b) is the selected row.

Overlap the selected row to the same row of the other parent(figure 2(c)).

Perform PMX

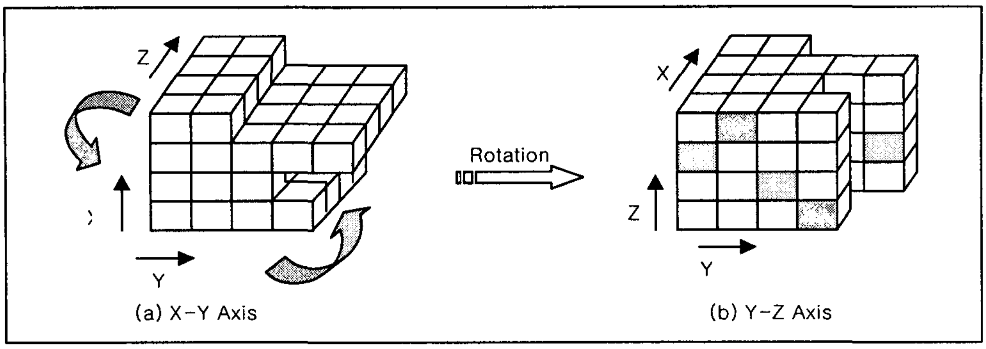
- The alleles, 'C','D','N', in figure 2(c) are redundant value.
- Replace 'C' of the row 4 with 'J' which is the original value of the parent 1 before overlapping.
- Replace 'D' of the row 4 with 'K' which is the original value of the parent 1 before overlapping.
- Replace 'N' of the row 2 with 'L' which is the original value of the parent 1 before overlapping.
- If there is no correspondent gene, delete that gene.

If weight exceeds the capacity of the vehicle, the last load(=gene) moves to the next vehicle(fig 2(e))

4) Local Search (Considering Y-Z axis (figure 3))

We rotate original chromosome into Y-Z axis. Every gene has the own time window. We should rearrange the order of gene in order to make a feasible route. To achieve local optimality, hill-climber algorithm is used. If the infeasible route happen, the gene move into next vehicle. If the feasible route happen, NNH(nearest neighbor heuristic) is used.

For preventing the infeasible solution, we add the penalty terms to the fitness function. That



(a) X-Y Axis (b) Y-Z Axis

<Figure 3> Chromosome Direction

is, the fitness of the chromosome has the terms with the sum of fitness of every vehicle, and penalties corresponding to the violating constraints. And there is no mutation in this paper.

5) Fitness

$$\mathbf{F(x)} = \sum_{i \in V} F_i + V * M$$

$$F_i = f_j + \{\max[0,\ a_j(f_{j-1}+d_{j-1,j})] + \alpha * \max[0,\ (f_{j-1}+d_{j-1,j})-b_j]\}$$

$$f_j = d_{j-1,j} + \{\max[0,\ a_j - (f_{j-1} + d_{j-1,j})] + \max[0,\ (f_{j-1}+d_{j-1,j}) - b_j]\} \quad , j \in N_i$$

V ; The set of vehicles. $d_{j-1,j}$; The time from (j-1)-th to j-th node.

a_j ; The earliest time for service of the j-th node.

b_j ; The latest time for service of the j-th node.

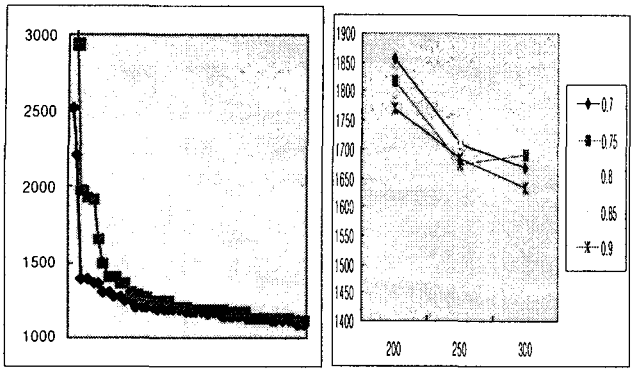
N_i ; The set of nodes in vehicle i. f_j ; Spending time from 0 to j-th node.

F_i ; The fitness of vehicle i. M, α ; Penalty.

The first term in F_i indicates moving time from depot to (j-1)th node. The second term means waiting time and the third term is lateness. The last term is added in order to reduce the number of vehicles. Penalty M and α are the arbitrary big constant.

4. Computation Experiments

The algorithm was coded in Visual C++ and the experiments were conducted on a Pentium II 266MHz. Problem difficulty increases with the time increment of the number of windows, because the rate of overlapping time windows increases. As noted by Dumas et al.[3], the running time of their exact algorithm increases exponentially with time window width for a given problem size. We simulated with the Solomon's data. In Solomon's instances[25], he changed the unit of distance into unit of time, since distance can be converted into time. Travel times between customers are truncated to one decimal place. Unfortunately, there are no published instances for VRPTWHtF. And we modified Solomon's problem(r101) and simulate it with various parameter values. Each node has time window [a, b]. Duration t_{ij} is associated with each arc(i, j). The VRPTW problems generated by Solomon[25] incorporate many distinguishing features of vehicle routing with two-sided

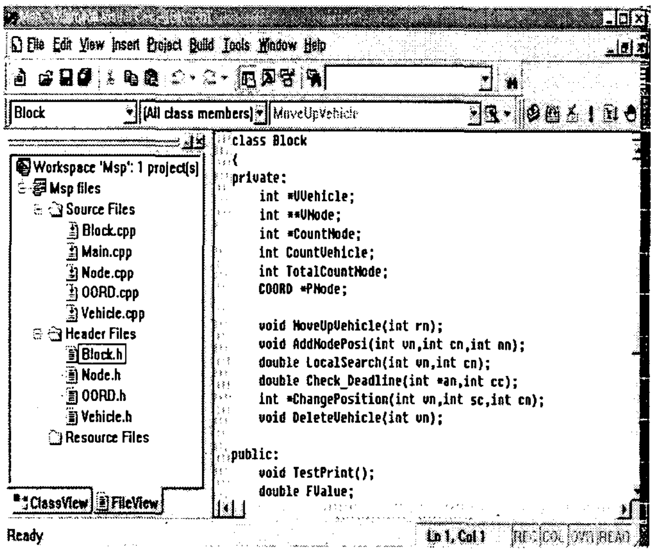


(a) (b)

<Figure 4> Result

<Table 1> Result

Capa -city	p_c / #Veh	0.70	0.75	0.80	0.85	0.90
CASE1 200 (#9)	1(200)	133	254	237	254	218
	2(200)	200	232	219	236	229
	3(200)	254	254	262	194	221
	4(200)	246	176	255	244	146
	5(200)	242	192	184	231	240
	6(200)	253	234	143	157	228
	7(200)	245	245	212	224	247
	8(200)	192	231	250	245	240
	9(200)	88				
Fitness		1855	1818	1761	1785	1771
CASE2 250(3) 200(4) 150(1)	1(250)	164	164	231	129	236
	2(250)	223	205	247	215	239
	3(250)	240	252	225	214	106
	4(200)	253	193	237	205	253
	5(200)	237	248	211	234	211
	6(200)	274	214	229	244	248
	7(200)	224	241	229	219	208
	8(150)	94	157	194	203	182
Fitness		1709	1673	1711	1664	1684
CASE3 300(4) 250(3) 200(1)	1(300)	250	279	225	215	236
	2(300)	235	251	179	238	214
	3(300)	221	224	243	240	240
	4(300)	249	204	238	220	192
	5(250)	276	227	243	225	243
	6(250)	219	269	218	245	225
	7(250)	187	235	170	130	219
	8(200)	30		48		64
Fitness		1667	1689	1564	1513	1633



```
class Block
{
private:
    int *VVehicle;
    int **VNode;
    int *CountNode;
    int CountVehicle;
    int TotalCountNode;
    COORD *PNode;

    void MoveUpVehicle(int rn);
    void AddNodePosi(int vn,int cn,int nn);
    double LocalSearch(int vn,int cn);
    double Check_Deadline(int *an,int cc);
    int *ChangePosition(int vn,int sc,int cn);
    void DeleteVehicle(int vn);

public:
    void TestPrint();
    double FValue;
```

<Figure 5> Program Demo

time windows. The problems vary in fleet size, vehicle capacity, travel time of vehicles, spatial and temporal distribution of customers, time window density(the number of demands with time window), time window width, percentage of time constrained customers and customer service times. In this example, time window is divided into morning and afternoon. For simulating GA, we let population size 50, iteration size 2000, p_m zero and penalty M and α 50. Generally, the more iteration we have, the better route is generated and the more population size we have, the better route is generated too. The result is given table 1 and figure 4(b). Figure 4(a) depict the improvement process of population's fitness and figure 5 is the demo of source program.

Each routes(rows) is generated from each vehicle. From table 1 and the figure 3, generally good routes is obtained at Pc = 0.8~0.9. Although the best route is generated at Pc = 0.85 in case 3, we should decide an alternative plan by considering the cost and time simultaneously.

5. Conclusion

This paper has described an application of genetic algorithm for vehicle routing problems. We found the reasonable set of routes for the testing problem. But the proposed approach is not competitive to the best heuristic technique for the VRPTWHtF. But, there are no VRPTW problems considering heterogeneous fleet using GA.

Cubic-GA algorithm has a weakness. When the time windows' width is narrow, the performance of the algorithm is inferior to the published best solution. Such a case we can guess that the component of local search may not be elaborate. But Cubic-GA is a quite flexible algorithm and we are able to find good routes whatever types of vehicles.

There are several issues for future research. First, we need a heuristic algorithm considering tight time windows. Second, alternatives of some components of this algorithm may be investigated(e.g. various crossover operators, and other local searches).

Referance

[1] Aykin; "Networking Policies for Hub-and Spoke Systems with Application to the Air Transportation System", Trans. Sci., Vol.29, No.3, pp.201 ~221, 1995.

[2] L.Davis; "Job shop scheduling with genetic algorithms", Proc. of the 1st Int.conf. Genetic Algorithms and Their Applications, Pittsburg, PA, pp.136~140, 1985.

[3] Y.Dumas, J.Desrosiers, E.Gelinas, and M.Solomon; "An Optimal Algorithm for the Traveling Salesman Problem with Time windows", Opns. Res., Vol.43, No.2, pp.367~371, 1995.

[4] M.Fisher, K.Jrnsten, and O.Madsen; "Vehicle Routing with Time Windows : Two Optimization Algorithms", Opns. Res., Vol.45, No.3, May-June, pp.488~492., 1997.

[5] B.Freisleben and P.Merz; "A Genetic Local Search Algorithm for Solving Symmetric and Asymmetric Traveling Salesman Problems", in Proceedings of the 1996 IEEE International Conference on Evolutionary Computation, Japan, pp.616~621, 1996.

[6] D.Goldberg; Genetic Algorithms in Search Optimization, and Machine Learning, Addison-Wesley, pp.166~179, 1989.

[7] D.Goldberg and Lingle; "Alleles, loci, and the traveling salesman problem", Proc. 1st Int. Conf. Genetic Algorithms and their Applications, pp.154~159, 1985.

[8] J.Grefentte, R.Gopal, B.Rosmatia, and D.Gucht; "Genetic Algorithms for the Traveling Salesman Problem", Proc. of the 1st International Conference on Genetic Algorithms and their Applications, pp.160~168, 1985.

[9] A.Homaifer, S.Guan, and G.Liepins; "A New Approach on the Traveling Salesman Problem", Proc. of the 5th Int. Conf. Genetic Algorithms, pp.460~466, 1993.

[10] A.Kahng and B.Moon; "Toward more powerful recombinations", Proc. of the 6th Int. Conf. Genetic Algorithms, pp.96~103, 1995.

[11] K.Katayama, M.Hirabayashi, and H.Narihisa; "Performance Analysis of a New Genetic Crossover for the Traveling Salesman Problem", IEICE Trans. Foundation., Vol.E81 No.5, pp.738~750, 1998.

[12] S.C.Lin, E.D.Goodman and W.F.Punch; "A Genetic Algorithm Approach to Dynamic Job Shop Scheduling Problems", Proc. of the 7th Int. Conf. Genetic Algorithms, pp.481~488, 1997.

[13] H.Matsuhashi; "Asynchronous Massively Parallel Genetic Algorithm Using Artificial Sociey Model and its Application", Proc, of the International Symposium on Optimization and Innovative Design, pp.226~229, 1997.

[14] P.Merz and B.Freisleben; "A Genetic Local Search Approach to the Quadratic Assignment Problem", Proc. of the 7th Int. Conf. Genetic Algorithms, pp.465~472, 1997.

[15] H.Muhlenbein, G.Schleuter, and Kramer; "Evolution Algorithms in Combinatorial Optimization", Parallel Computing 7, North-Holland, pp. 65~85, 1988.

[16] T.Murata, H.Ishibuchi, and M.Gen; "Neighborhood Structures for Genetic Local Search Algorithms", Proc. of the 2nd Int. Conf. On Knowledge-Based Intelligent Eletronic Systems, Vol.2, pp. 259~263, 1998.

[17] K.Ohkura and K.Ueda; "An Extended Framework for Overcoming Premature Convergence", Proc. of the 7th Int. Conf. Genetic Algorithms, pp.260~267, 1997.

[18] Oliver, Smith, and Holland; "A study of permutation crossover operators on the traveling salesman problem", Proc. of the 2nd Int. Conf, Genetic Algorithms and Their Applications, pp.224~230, 1987.

[19] P.Prinetto, M.Rebaudengo and M.Reorda; "Hybrid Genetic Algorithms for the Traveling Salesman Problem", Artificial Neural Nets and Genetic Algorithm Proceedings of the International Conference in Innsbruck, Austria, pp. 559~566, 1993.

[20] T.Starkweather, S.McDaniel, K.Mathias, D.Whitley, and C.Whitley; "A comparison of genetic sequencing operators", Proc. of the 4th Int. Conf. Genetic Algorithms, pp.69~76, 1991.

[21] S.Thangiah, "An Adaptive clustering Method using a Genetic Shape for Vehicle Routing Problems with Time Windows", Proc. 6th Int. Conf. Genetic Algorithms, pp.536~543,1995.

[22] D.Whitley, T.Starkweather, and D.Fuguay; "Scheduling Problems and Traveling Salesman : The Genetic Edge Recombination Operator", Proc. of the 3rd Int. Conf. Genetic Algorithms, pp.133~140, 1989.

[23] M.Yamamura, I.Ono, and S.Kobayashi, "Character-preserving Genetic Algorithms for Traveling Salesman Problem", Journal of Japanese society for Artificial Intelligence, Vol.7, No.6, pp.1049~1059, 1992.

[24] Y.S.Yoo and I.G.Lo; "Vehicle Routing Problem Using Parallel Genetic Algorithm", Proc. of the 18th Korea Industrial Engineering Fall Conference, pp1179~1183, 1998

[25] http://dmawww.epfl.ch/~rochat/rochat_data/solomon.htm