A vehicle routing problem solved by Agents

Mª Belén Vaquerizo García
Languages and Systems Area
Burgos University, Burgos
belvagar@ubu.es

Abstract. The main purpose of this study is to find out a good solution to the vehicle routing problem considering heterogeneous vehicles.

This problem tries to solve the generation of paths and the assignment of buses on these routes. The objective of this problem is to minimize the number of vehicles required and to maximize the number of demands transported.

This paper considers a Memetic Algorithm for the vehicle routing problem with heterogeneous fleet for any transport problem between many origins and many destinations. A Memetic Algorithm always maintains a population of different solutions to the problem, each of which operates as an agent. These agents interact between themselves within a framework of competition and cooperation.

Extensive computational tests on some instances taken from the literature reveal the effectiveness of the proposed algorithm.

Keywords: Vehicle Routing Problem, Heterogeneous Fleet, Evolutionary Algorithms, Memetic Algorithms, Agents

1 Introduction

A vehicle routing problem with heterogeneous fleet is one of the most important problems in distribution and transportation. So, this generic problem and its practical extensions are discussed in great detail in the literature [2].

This paper considers a Memetic Algorithm for a vehicle routing problem with heterogeneous fleet for any transport problem between many origins and many destinations. The objective of this problem is to minimize the number of vehicles required and to maximize the number of demands transported.

A good example of this type of problem is the urban and interurban transport. In this problem is considered the heterogeneous fleet and networks symmetric or asymmetric. In the heterogeneous fleet, different types of vehicle with different capacities may be considered, and, furthermore, only in symmetric networks, the distance between two nodes is considered as the same in two directions.

More precisely, this problem can be defined as a problem of determining a set of routes and the best assignment of the available vehicles over these routes. And, besides, this problem must cover all passenger demand subject to the next constraints:

- Each vehicle performs just one route.
- For each route, the total number of passengers to be covered in each period of time considered, should not exceed the maximum capacity of the transport vehicle, so

it is necessary to determine the frequency of vehicles to take this route in each time period.

- The total transportation cost should be minimized.
- Tours with setbacks are not allowed on routes.

According to this, the planning of this problem is as it is shown in the next figure.

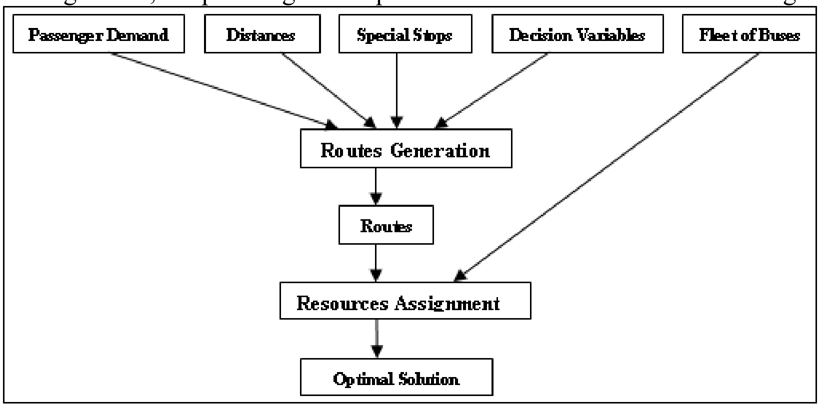

Fig 1. Planning of an Urban Public Transport

By other hand, in the theory of computational complexity, this type of problem belongs to the class of NP-complete problems. Thus, it is assumed that there is no efficient algorithm for solving it, and in the worst case the running time for an algorithm for this problem depends exponentially on the number of stops, so that even some instances with only dozens of stops cannot be solved exactly.

In artificial intelligence, evolutionary algorithms are a large set of stochastic optimization algorithms inspired by biology [1]. They are based on biological processes that allow populations of organisms adapt to their environment. Also they can be considered for searching solutions.

Memetic Algorithms belong to the class of evolutionary algorithms that hybridize local search within a classical Genetic Algorithm framework. They are an optimization paradigm based on the systematic knowledge about the problem to be solved, and the combination of ideas taken from different metaheuristics, both based on population as based on local search. It is an Algorithm that can be viewed as a variant of Genetic Algorithms. Basically, they combine local search heuristics with crossover operators. For this reason, some researchers have viewed them as Hybrid Genetic Algorithms [12]. However, combinations with constructive heuristics or exact methods may also belong to this class of metaheuristics.

A Memetic Algorithm always maintains a population of different solutions to the problem, each of which operates as an agent. These agents interact between themselves within a framework of competition and cooperation. It is very similar to what occurs in nature between individuals of the same species. Each generation consists of updating the population of agents [3].

Otherwise, it is necessary to adapt the algorithm to the characteristics of the problem to be solved [7]. Thus, it needs to use a guide function that quantifies how good are the agents to can solve this problem.

In conclusion, the framework of this research is the development of and effective metaheuristic for hard combinatorial optimization problems met in vehicle routing [9].

Therefore, this paper tries to solve, concretely, the planning in the collective urban public transport, and it is structured as follows: In Section 2 this paper analyzes the Memetic Algorithm to be used to solve this problem. Section 3 describes the main features and the mathematical model of this problem to be considered in its resolution. And, Section 4 presents the construction of routes and the assignment of the buses to them, both of them developed through a Memetic Algorithm. This paper finalizes with the conclusions and evaluation of the strategy used.

2. Evolutionary Algorithm. Memetic Algorithm

The basis of this algorithm is a set or population of solutions on which some operations are performed, such as selection of parents, crossover, reproduction, selection of survivors or mutation, and each of the iterations in which this sequence is performed, are called generation [4].

Both, the selection and the replacement processes, are purely competitive, and in they only vary the distribution of the existing agents. The reproduction of agents is usually done through two operators: recombination and mutation.

Thus, the main feature of this algorithm is the use of crossover or recombination operator as the primary mechanism for search, and to build descendants of chromosomes that have the same features as the chromosomes that are crossed. Its usefulness is the assumption that different parts of the optimal solution can be discovered independently and, then, they can be combined to form better solutions. It ends when a certain number of generations without improvement are passed, or when other predetermined criteria are met.

By other hand, the basic idea of this algorithm is to incorporate as much knowledge to the problem domain as it is possible during the process of generating a new population. In order to this, this knowledge can be incorporated in the selection of the attributes of the parents to be transmitted to children, and in the selection of the attributes that are not parents to be transmitted to children.
Although in a local search algorithm it was defined a neighbourhood for a single individual, in a population of individuals the vicinity of that population can be obtained through the composition of the individuals.

However, the objective is to build new solutions from the existing ones, and this can be done by identifying and combining attributes of the current solutions.

Otherwise, this algorithm requires two things to be defined: A genetic representation of the solution domain, and a fitness function to evaluate the solution domain.

According to this, the framework of the proposed Memetic Algorithm used in this work is the next [11]:

```
Procedure MA;
{
   for (j=1;j++; j<=popsize)
   {
      i=GenerateSolution();
      i=Local-Search(i);
      add individual i to P;
   }
   while terminate!=true
   {
        for (i=1;i++;i<= #recombination)
        {
             select two parent i_a,i_b ∈ P Randomly;
             i_c =Recombine(i_a,i_b);
             i_c =Local-Search(i_c);
             add individual i_c to P;
        }
        for (i=1;i++;i<= #mutations)
        {
             select an individual i∈P Randomly;
             i_m=Mutate(i);
             i_m=Local-Search(i_m);
             add individual i_m to P;
        }
      P=select(P);
      if (P converged) P=mutateAndLS(P);
   }
}
```

Fig 2. Framework of the Memetic Algorithm used

3. The Vehicle Routing Problem. Public Transport Routes

As mentioned above, this work is done to try to improve the urban public transport in any mid-size city, and, for that, previously has to be analyzed the current situation in that city, in order to obtain a detailed knowledge of its structure and function with regard to the following points:

- The number of lines and characteristics of their stops.
- The frequency of each line.
- The estimated speed at which buses can go through the city.
- The planning schedules of the lines.
- The estimation of the passengers getting on and off at each stop and in each time period considered within the timeframe set.
- The number of buses available. Different types of buses available and their restrictions.
- Special passengers demand in specific stops and in specific periods of time.
- The network topography of the city in order to know the limitations provided by the city streets as a cut street, a narrow street, a one-way street, etc.

Therefore, the identification of all these data (input data and decision variables) will allow subsequently to formalize the problem to solve.

As discussed before, this problem is considered a complex problem of combinatorial optimization, which belongs to the class of problems known as NP-hard, those whose exact resolution requires specific models and algorithms with great difficulty and much processing time to obtain the optimal solution. For this reason, when a problem about routes is discussed, it is often addressed through the formulation of models whose decisions can be obtained through algorithms to solve instances of suitable size in an optimal way and in a reduced time.

By other hand, it should be noted that the evaluation of the solution implies calculating the objective function of the model, and that, moreover, it implies to choose a resolution method, in which there must be a commitment between efficiency and quality of the solution obtained.

Thus, the mathematical model, that represents the problem studied, can be seen as a single objective function (maximize the level of service, considering the number of buses available as a restriction). In this case, it should be noted that the approach to maximize the level of service is considered as minimizing travel time and waiting time of passengers at stops. In the next table, the mathematical model, for the geographical network of a city to be considered, is presented detailing the notation as follows:

$G = (A, N)$ where $N = \{1, 2,...,n\}$ (*set of nodes*) and $A = \{a1, a2 ...,am\}$ (*set of arcs*)

$P \subset N$, $P = \{i1, i2,..., ip\}$ (*set of stops*) $\forall\ i, i' \in P$ *demii' = Passenger demand from stop i to stop i'*

RT = {RR1, RR2,... RRs, ...,RRntrc} set of Full Routes

σ(i, i') = {Routes with this pair of stops in their path} *Xs* = Number of vehicles assigned to the route *RRs*

tpii',s = Travel time from the stop *i* to the stop *i'* in the Route *RRs*

ntv = Number of different types of vehicles *λ(l) = {Routes covered by vehicles of the type l}*

freqs = *number of times you can start the route s in that period of time*

The Objective Function would be to Minimize Waiting Times and Travel Times, with the restrictions (3, 4) :

(1) Waiting Times at Stops:

$$\sum_{i,i' \in P} \frac{60}{\sum_{s \in \sigma(i,i')} freq_s} \cdot dem_{i,i'}$$

(2) Travel Times:

$$\sum_{i,i' \in P} \left[\left(\sum_{s \in \sigma(i,i')} tp_{ii',s} \cdot freq_s \right) \Big/ \sum_{s \in \sigma(i,i')} freq_s \right] \cdot dem_{ii'}$$

(3) Number of Vehicles for each type: $\forall\ l = 1... ntv$

$$\sum_{s \in \lambda(l)} X_s \le nv_l$$

(4) All demand covered: $\forall\ i, i' \in P\ /\ dem_{i,i'} > 0$

$$\sum_{s \in \sigma(i,i')} X_s > 0$$

Table 1. Mathematical Model for the Planning of a Urban Public Transport

4. Solving the problem using a Memetic Algorithm

This work has implemented the following general layout of a Memetic Algorithm: Once the genetic representation and the fitness function are defined, the algorithm proceeds to initialize a population of solutions randomly, then it is improved through a repetitive application of mutation, crossover, inversion and selection operators [5,6,10].

The evolution starts from a population of randomly generated individuals and happens in generations. In each generation, the fitness of every individual in the population is evaluated, multiple individuals are stochastically selected from the current population (based on their fitness), and modified (recombined and possibly randomly mutated) to form a new population. The new population is then used in the next iteration of the algorithm. Thus, each generation consists on updating the population of agents, using for it a new population obtained by recombination of the characteristics of some selected agents [13]. The selection operation choose a selection of the best agents in the current population, it requires the use of a function for measuring the quality of each agent in the resolution of the problem. Then, the competition between agents is through the selection and update operations.

This algorithm terminates when either a maximum number of generations has been produced, or a satisfactory fitness level has been reached for the population. If the algorithm has terminated due to a maximum number of generations, a satisfactory solution may or may not have been reached.

By other hand, is very important to fix some features in the representation of the objective functions, which are efficiently exploited by a memetic approach. The first element to be determined is the representation of solutions to be used. It is important to clarify here that the representation should not be built as merely coding, for which the relevant considerations are related to memory consumption, complexity handling, etc. On the contrary, the representation refers to the abstract formulation of the solutions from the perspective of the algorithm. In this case, arrays of different structures have been used, and in this problem it has been determined that the most appropriate way to store the information have been by adjacency.

Besides, the fitness function is defined over the genetic representation and measures the quality of the represented solution. Candidate solutions to the optimization problem play the role of individuals in a population, and the fitness function determines the environment within which the solutions are considered valid.

The computational evaluation was performed on little instances of about 60 stops. Thus, considering the variables and restrictions listed before and for the following network, as an example in Figure 3, the system would provide the next designs of routes shown below in Table 2.

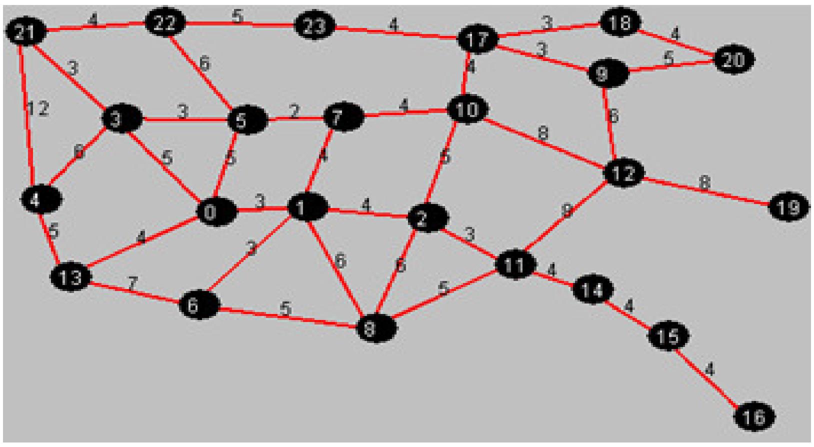

Fig 3. Topographical network considered as example. Stops, arcs and distances

Routes	Path	Demand covered	Travel Time	Buses
1:				3,7
→:	20 18 17 10 7 5 0 13	41	26	
←:	13 0 1 6 8 2 10 17 18 20	81	37	
2:				4,6,8
→:	16 15 14 11 2 1 7 5 22	71	31	
←:	22 21 3 0 1 2 11 14 15 16	107	34	
3:				1
→:	4 13 6 1 7 10 17 9	43	30	
←:	9 17 10 7 5 3 4	23	22	
4:				5
→:	4 3 5 7 10 12 19	31	31	
←:	19 12 10 7 5 3 4	29	31	
5:				2
→:	8 11 12 9 17 23 22	37	31	
←:	22 21 3 0 1 8	22	21	

Table 2. A solution about generation of routes and their buses using a Memetic Algorithm

These feasible routes satisfy specific conditions based on the number of stops, total distance, maximum time travel, passenger demand satisfied, etc.

To obtain this solution, it has been necessary to define in the algorithm the following points, among others: Number of individuals to cross, how selecting the parents, the type of selection, the type of combination, the type of crossing, the number of children.

Moreover, in this problem, routes are going to be exchanged and the individuals which are going to be crossed are solutions of the current population. And new individuals called children are obtained by crossing. The value of the solution, it is used to sort the solutions in the population to be the best solution which holds less value. This value is based on the ratio between the demand and travel time of the solution.

Conclusions and future lines

The computational experiments have been done on some of the instances available in the TSPLIB.The results obtained have been contrasted with the results obtained with a Genetic Algorithm for the same instances of the same problems, providing a better solution with the Memetic Algorithm because it has covered a greater demand and has reduced travel time on routes.

Moreover, this algorithm can be improved with different local search algorithms, to improve the quality of solutions. These local search methods will make use of explicit strategy parameters to guide the search, and will adapt these parameters with the purpose of producing more effective solutions [8]. By other hand, also it is proposed, as future lines, the application of a new evolution strategy based on clustering and local search scheme for some kind of large-scale problems of this type [14], and the combined use of two jobs whose results obtained have been effective [2,11].

In addition to this, it can be quite effective to combine this algorithm, Memetic Algorithm, with other optimization techniques, such as simple Hill Climbing, Scatter Search, that are quite efficient at finding absolute optimum in a limited region.

Furthermore, the traditional studies of this type of problem assume that travel speed of the vehicle is constant, which is an approximation of the real-world conditions in order to simplify the model of computing, but in the real-world this assumption is not very reasonable, especially in urban city conditions. By this reason, it is ssugested in a future work to consider the study on time-dependent vehicle routing problems in regard to congestion during peak time in urban areas, accidents like vehicle breakdown, severe weather factors and so on.

Finally, among the main difficulties of this work, to get the service previously established, is the need to know the number of people using each line for each time period. For these reasons, for a better treatment of this problem, it is suggested in a future work the use of the Fuzzy Sets Theory, and the combination of a Memetic Algorithm with others efficient algorithms such as mentioned above.

References

1. Alba, E., Cotta, C., Herrera, F. Computación Evolutiva. Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial, 1998.
2. Buriol, L., Franca, P.M., Moscato, P. A new memetic algorithm for the asymmetric traveling salesman problem. Journal of Heuristics, Volume 10. 2004.
3. Carrascosa, C., Bajo, J., Julian V., Corchado, J.ML, Botti, V.J. Hybrid multi-agent architecture as a real-time problem-solving model. Expert Syst. Appl. (ESWA) 34(1):2-17. 2008.
4. Cotta, C. Una visión general de los algoritmos meméticos, Rect@ 3:139-166. 2007.
5. García, S., Cano, J.R., Herrera, J.R. A memetic algorithm for evolutionary prototype selection: A scaling up approach. Pattern Recognition (PR) 41(8):2693-2709. 2008.
6. Gutin, G., Karapetyan, D. A Memetic Algorithm for the Generalized Traveling Salesman Problem. CoRR abs/0804.0722. 2008.
7. Handa, H., Chapman, L, Yao, X. Robust Salting Route Optimization Using Evolutionary Algorithms. Evolutionary Computation in Dynamic and Uncertain Environments. 2007.
8. Molina, D., Lozano, M., García-Martínez, C., Herrera, C. Memetic Algorithm for Intense Local Search Methods Using Local Search Chains. Hybrid Metaheuristics 2008:58-71.
9. Prins, C. A simple and effective evolutionary algorithm for the vehicle routing problem. Computers and Operations Research. 2004.
10. Rodrigues, A.M., Soeiro Ferreira, J. Solving the rural postman problem by memetic algorithms. In Jorge Pinho de Sousa, editor, Proceedings of the 4th Metaheuristic International Conference (MIC'2001), Porto, Portugal. 2001.
11. Tavakkoli-Moghaddam, R., Saremi, A. R., Ziaee, M. S. A memetic algorithm for a vehicle routing problem with backhauls. Applied Mathematics and Computation, Volumen 181. 2006.
12. Wei, P., Cheng, L.X. A hybrid genetic algorithm for function optimization. Journal of Software, 10(8):819{823, 1999.
13. Xu Jin, Habib Abdulrab, Mhamed Itmi: A multi-agent based model for urban demand-responsive passenger transport services. IJCNN 2008:3668-3675.
14. Yuping Wang, Jinhua Qin: A Memetic-Clustering-Based Evolution Strategy for Traveling Salesman Problems. RSKT. 2007.