The Multi-Swarm Particle Swarm Optimization Algorithm for Vehicle Routing Problem

Du Taotao
Shanghai University of Engineering Science
School of Management
Shanghai, China.

Liu Sheng
Shanghai University of Engineering Science
School of Management
Shanghai, China.

Abstract

Aiming at the precocious convergence problem of particle swam optimization, multi-swarm particle swarm optimization algorithm based on particle evolution is presented. The algorithm uses local version of the particle swarm optimization method, from "particle evolution" and "multiple groups" aspects to improve standard particle swarm algorithms. Multiple particle swarm searches solution space independently which keeps diversities of particle populations and improves the stability of the algorithm effectively. The experimental results show that the new algorithm has advantages of convergence property, convergence speed, accuracy and the stability of convergence effective

Keywords: Vehicle Routing Problem, Multi-swarm, particle swarm optimization algorithm

1. Introduction

Nowadays the modern logistics has been recognized as the third important source of enterprises to create profits besides reducing material consumption and improve labor productivity, as well as the important way to reduce the cost of production and operation and to enhance the competitiveness of product markets. As a most important part of the logistics system, logistics plays a key role on the efficiency of the whole logistics system. Vehicle scheduling is one of the most important logistics management. Vehicle routing problem (VRP) is an important research direction of logistics, which are also a typical vehicle scheduling problem and also an important combinatorial optimization problem.

2.VRP

A. The origin of the VRP

VRP is proposed firstly by Dantzig and Ramser in 1959.It refers to a certain number of customers who have different quantity of demand of goods, at the same time goods distribution center to provide customers by a motorcade and organize the appropriate route. And the goal is to make customers satisfied, and to achieve some purpose of the shortest distance, the smallest cost and least time consuming under certain constraints.

As a NP - hard problem, VRP problem is difficult to find the exact solution. Since proposed and development up to now, a lot of research was carried out and design various types of algorithm. Many scholars have carried on the induction and classification of these methods which can basically be divided into two categories: precise model and the heuristic. Among them, the precise type algorithm generally can be divided into tree search method, branch and bound method, dynamic programming and an integer linear programming etc; Heuristic algorithm can be divided into two stage approach, constructive algorithm and the incomplete type algorithm such as simulated annealing algorithm, tabu search method, genetic algorithm and ant colony algorithm etc. These methods are also known as the heuristic algorithm in some categories.

B.VRP Model

In order to simplify the model to analyze and illustrate this problem, VRP model are defined as follows:

(1)Only one central warehouse, every car starting from here, and send the goods back here;
(2)The distance of the center warehouses and various service point is the distance of two points coordinates;
(3)All vehicle load is the same;
(4)Every client only needs one vehicle;
(5)Not affected by objective factors such as weather and vehicle failure ;

Based on the above assumptions, parameter symbols and decision-making variables are expressed as follows:

$q_k, (k = 1,2,\cdots,K)$: Car's bearing capacity and have the number of available cars is K;

$g_i, (i = 1,2,\cdots,L)$: The service point i demand of the goods and have the number of service points is L;

c_{ij}: Transportation cost from i to j;

$$x_{ijk} = \begin{cases} 1, \textit{Vehicle} \ \mathrm{K} \ \textit{from } i \text{ to } j \\ 0, \textit{otherwise} \end{cases}$$

$$y_{ik} = \begin{cases} 1, \textit{Task of servicepoint } i \textit{ are } \text{performed } \textit{by vehicle } \mathrm{K} \\ 0, \textit{otherwise} \end{cases}$$

And $\max \mathrm{g_i} \le \max q_k$ We can get the mathematical model of VRP as follows:

$$\min C = \sum_{i=0}^{L}\sum_{j=0}^{L}\sum_{K=1}^{K} c_{ij} x_{ijk} \qquad (1)$$

St. $$\sum_{i=1}^{L} g_i y_{ik} \le q, k = 1.2.\cdots,K \qquad (2)$$

$$\sum_{k=1}^{K} y_{ik} = \begin{cases} 1, i=1,2,\cdots,L \\ K, 1=0 \end{cases} \qquad (3)$$

$$\sum_{i=0}^{L} x_{ijk} = y_{jk}, j = 1,2,\cdots,L; k = 1,2,\cdots,K \qquad (4)$$

$$\sum_{j=0}^{L} x_{ijk} = y_{jk}, j = 1,2,\cdots,L; k = 1,2,\cdots,K \qquad (5)$$

$$x_{ijk} = \{0,1\}, j = 1,2,\cdots,L; k = 1,2,\cdots,K \qquad (6)$$

$$y_{jk} = \{0,1\}, j = 1,2,\cdots,L; k = 1,2,\cdots,K \qquad (7)$$

Among these, the objective function (1) guarantees the total cost C minimum;
Constraint (2) expresses the load is less than or equal to their maximum value;
Constraint (3) denotes that every service points only need one vehicle and the task of all service points are performed by K cars together;
Constraint (4) and (5) show that every service points only needs one vehicle;
Constraint (6) and (5) guarantees decision-making variables is a 0-1 variable.

III. Introduction of Pso

A. Overview of PSO

PSO is proposed by Eberhart and Kennedy in 1995, it is the result of simulation of birds' predatory behavior. In this algorithm, each solution of optimization problem is surfing a bird in the search space and called a "particle"; the particle represents a potential solution, all of the particles have a fitness which is determined by the function of optimization. Each particle has a fly speed which determines their direction and distance. Then, Particle is searching follow the optimal particle groups. Iteration rules of standard PSO is as follow:

Calculate the new velocity and position of every particle according to function as follow:

$$V_{\mathrm{id}}(t+1)=V_{id}(t)+c_1r_1[p_{id}(t)-X_{id}(t)]+c_2r_2[p_{gd}(t)-X_{id}(t) \quad (2.1)$$

$$V_{\mathrm{id}}(t+1)=X_{id}(t)+V_{id}(t+1), \quad (2.2)$$

Among them t refers to iterations; $V_{id}(t)$ denotes particle velocity; $X_{id}(t)$ denotes the particle's position; $P_{id}(t)$ means ever best position of each particle; $P_{gd}(t)$ for the particle swarm optimal position; c1 and c2 are position constants; $r_1,r_2\in[1,0]$ are random number which belongs to 0-1.

Due to initial velocity formulas are very hard or even unable converged to global optimal, so Shi Y and other researchers introduced the inertia weight w to type (11), the formula becomes:

$$V_{\mathrm{id}}(t+1)=\omega V_{id}(t)+c_1r_1[p_{id}(t)-X_{id}(t)]+c_2r_2[p_{gd}(t)-X_{id}(t) \quad (2.3)$$

$$\omega=\omega_{\max}-(\omega_{\max}-\omega_{\min})\times\frac{iter}{iter_{\max}} \quad (2.4)$$

B. Coding

It is one of key issues in the realization of algorithm through finding a suitable expression to make particles and corresponding solution. We construct a 2L-dimension space corresponding to vehicle routing problem, and with L service points. Each service point corresponds to two dimensions: numbered the task vehicle is k, numbered the task's order of k's drive path is r. For the convenience and computation, each particle corresponding to 2L-dimensional vector X divided into two L-dimensional vectors: X_v (Represent each task and corresponding vehicles), X_r (Represents task's execution sequences in corresponding vehicle routings).

The biggest feature of the said method is that each delivery point can get vehicle delivery service and restrict each delivery complish only by a vehicle, so calculations are greatly reduced. Although the representation method has high dimension, but thanks to PSO has very good properties in algorithm optimization problem of multidimensional, computational complexity didn't increase along with dimension. We can see in the test node.

On the other hand, the set of initial solutions have a great impact on combinatorial optimization problems. Under normal circumstances, customers' code are represented by natural numbers, the natural number i indicates the i client (0 represents the warehouse). But we adopt a new code representation in my paper, for example, namely for K clients, m cars VRP problem, input (m-1) 0, divid customer sequences into m segments, each segment represents a walking paths. Each particle corresponding to (k+m-1) dimensional vector, that is the solution of problem.

For example: Suppose city- task are 7 and service vehicles are 3 for VRP problem. If a particle position vector is X, the serial number of service points:1,2,3,4,5,6,7;

X_v: 1 1 2 2 2 3 3 X_r: 1 4 3 1 2 2 1

Then the particle homographic solution path is following:

L1:0→1→2→0 L2:0→4→5→3→0 L3:0→7→6→0

The particle velocity vector V corresponding to X_v and X_r. The said method is that each delivery point can get vehicle delivery service and restrict each delivery complish only by a vehicle, so calculations are greatly reduced.

C. Process of PSO

a. Process of the Basic Algorithm

Basic algorithm process is as follow:

Table 1 Process of PSO

Start

Select the threshold ε and maximum iterations N_{max}

The location of initial particles: $Z_j^0 = (Z_{j1}, Z_{j2}, \cdots, Z_{jD}), j = 1,2,\cdots,m$

The initial velocity of each particle : $V_j^0 = (V_{j1}, V_{j2}, \cdots, V_{jD}), j = 1,2,\cdots,m$

Measure the fitness $Z_j^{(0)}$ of each particle ,Expressed as $\tilde{D}_j^{(0)}$ （Assume that solving the minimum value of the objective function）

$p_j^{(0)} = Z_j^0$

According to $\tilde{D}^{(0)} = \min\{\tilde{D}_1^{(0)}, \tilde{D}_2^{(0)}, \cdots, \tilde{D}_m^{(0)}\}$ to find the global optimal $P_g^{(0)}$

k=0

Step a k←k=1

According to (2.3) update V_i^k; According to (2.2) update Z_j^k

Step b Measuring Z_j adaptive value, Expressed as $\tilde{D}_j^{(0)}$

Update $P_i^{(k)}$ and $P_g^{(k)}$

If $(\tilde{D}^{(k-1)} - \tilde{D}^{(k)}) / \tilde{D}^{(k)} > \varepsilon$ and $k > N_{max}$,then go to step a

End

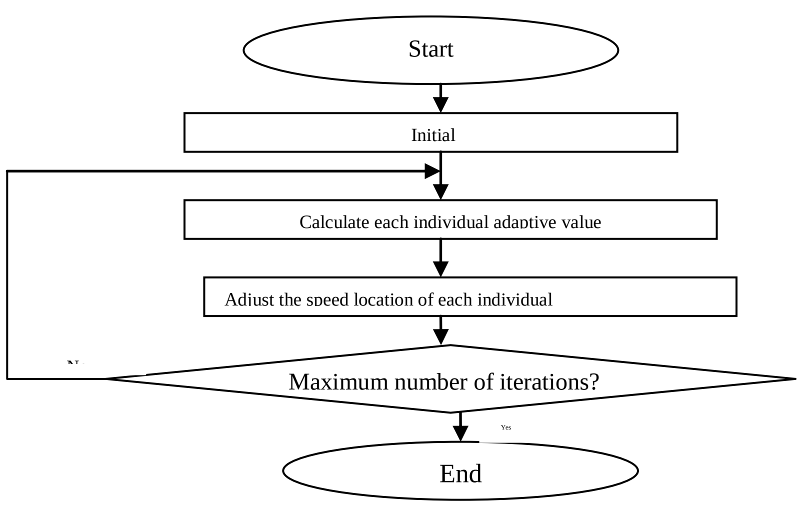


Fig 1: The Flow Chart of PSO

B. Improved Calculation Steps of the Pso Algorithm

The PSO algorithm is continuous space algorithm, and the VRP problem is an integer programming problem, so we should make the corresponding revision in the process of implementing. We will improve particle swarm optimization algorithm for multi particle swarm optimization algorithm based on particle evolution, the specific implementation steps are as follows:

Step 1 Initialize particle swarm

(1)Make particle swarm divid into a plurality of two two overlapping subgroups;
(2) Each particle position vector in each dimension were 1 ~ k (number of vehicles) between integers, each one-dimensional random takes 1 ~ l (shipping point between the real tasks number);
(3) Each velocity vector in each dimension random takes (k-1) ~ (k-1) (number of vehicles) between integers, each dimension of random takes number (l-1) ~ (l-1);
(4) Evaluated all particles with function Eval;
(5) Regard the initial evaluation as the individual optimal solution, and find the optimal solution and the total population in the subgroup within the optimal solution.

Step2 until a termination condition is satisfied, or get the maximum number of iterations.

(1) For each particle, according to the PSO distance, to update the velocity updating formula, exceed the range according to boundary value.
(2) Evaluation of all particles using evaluation function Eval.
(3) If the evaluation of a particle is better than the best rated value, the value is written into the history optimal evaluation value, meanwhile the current position is the position of the particle historical best position.
(4) Find the optimal solution and subgroups, if it is better than have given optimal solution, the history will be updated. For the subgroup of all individual even infeasible solutions, or in the subgroup with multiple individuals with the optimal solution, then randomly take one for subgroups within the current optimal solution.

IV. Experimental Analyses

In order to analyze the performance of particle swarm algorithm in the solution of VRP, we selected Shanghai, Beijing, Tianjin, Hongkong, Guangzhou, Zhuhai, Shenzhen and Hangzhou 8 city as an example, Shanghai is the central warehouse, the remaining are 7 service points, i.e. l=7; and 3 cars available, namely k=3 ; vehicle carrying capacity q1=q2=q3=1; the demands for each service point respectively g1=0.89, g2=0.14, g3=0.28, g4=0.33, g5=0.21, g6=0.41,g7=0.57;

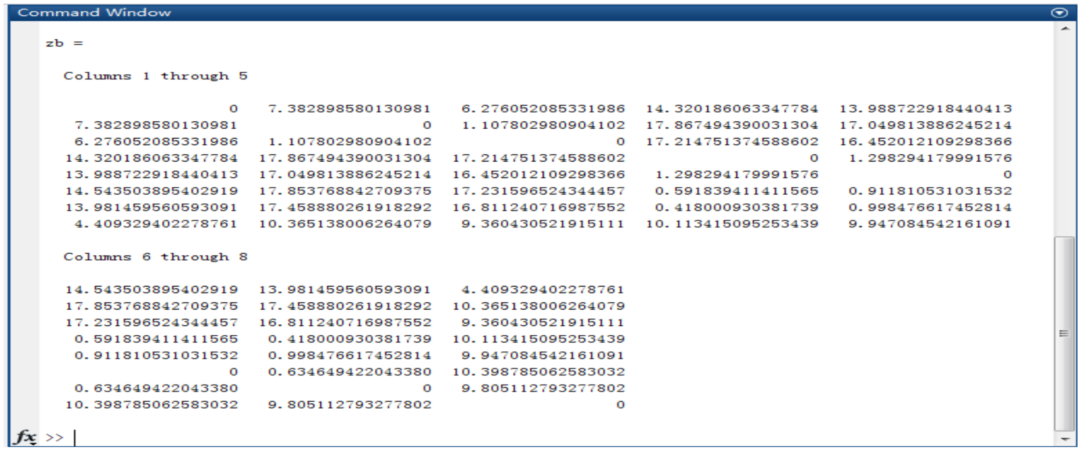
Table2: Fabric Coordinates of Each City

Cities	Latitude and longitude coordinates
Shanghai	(121.43333,34.50000)
Beijing	(116.41667,39.91667)
Tianjin	(117.20000,39.13333)
Hong Kong	(114.10000,22.20000)
Guangzhou	(113.23333,23.16667)
Zhuhai	(113.51667,22.30000)
Shenzhen	(114.06667,22.61667)
Hangzhou	(120.20000,30.26667)

(Source: Baidu library—the latitude and longitude coordinates of major cities in Chinese)

A. The Distance between the Center City and Each Service Points

Using matlab2008 calculates the distance between the center city and each city service points as shown:



```
Command Window

zb =

  Columns 1 through 5

                  0   7.382898580130981   6.276052085331986  14.320186063347784  13.988722918440413
  7.382898580130981                   0   1.107802980904102  17.867494390031304  17.049813886245214
  6.276052085331986   1.107802980904102                   0  17.214751374588602  16.452012109298366
 14.320186063347784  17.867494390031304  17.214751374588602                   0   1.298294179991576
 13.988722918440413  17.049813886245214  16.452012109298366   1.298294179991576                   0
 14.543503895402919  17.853768842709375  17.231596524344457   0.591839411411565   0.911810531031532
 13.981459560593091  17.458880261918292  16.811240716987552   0.418000930381739   0.998476617452814
  4.409329402278761  10.365138006264079   9.360430521915111  10.113415095253439   9.947084542161091

  Columns 6 through 8

 14.543503895402919  13.981459560593091   4.409329402278761
 17.853768842709375  17.458880261918292  10.365138006264079
 17.231596524344457  16.811240716987552   9.360430521915111
  0.591839411411565   0.418000930381739  10.113415095253439
  0.911810531031532   0.998476617452814   9.947084542161091
                  0   0.634649422043380  10.398785062583032
  0.634649422043380                   0   9.805112793277802
 10.398785062583032   9.805112793277802                   0

fx >> |
```

Fig 2: The Distance between the Center City and Each Service Points

B. Solution of VRP by PSO

In order to obtain more preciser optimal solution, initialize group n=40; inertia weight w=0.7298; $c_1 = c_2$ =1.4962;maximum iteration number MaxDT=50;Search space dimensions D=7;Experiment is expected to 50 iterative times which conducted a total of 20operations. The experimental results are described based on 20 repeat operation statistics in the following table:

Table3 Result Of the Operation 20 Times

Times	1	2	3	4	5	6	7	8	9	10
Total distance	53.477	53.477	57.443	53.477	57.443	53.477	53.477	53.477	53.477	53.477
Times	11	12	13	14	15	16	17	18	19	20
Total distance	53.477	72.475	53.477	53.477	53.477	53.477	53.477	53.477	53.477	53.477

From analysis of experimental results, the optimal route as follows: 0→1→2→0→3→4→5→6→0→7→0

Corresponding route is as follows:

Table4 Vehicle's Path

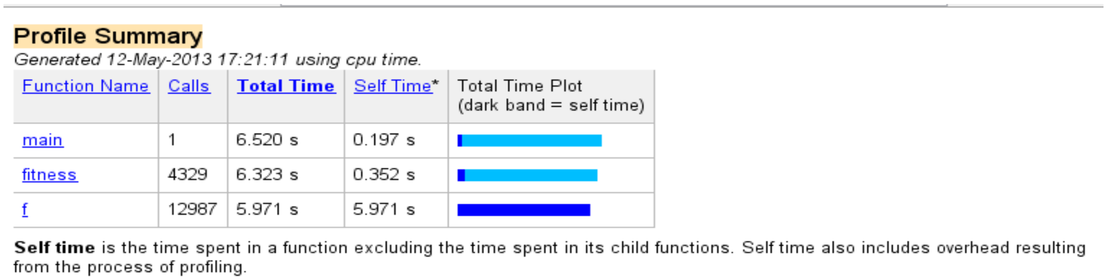
Vehicle k	Driving path
1	0→1→2→0
2	0→3→4→5→6→0
3	0→7→0

Total driving distance is 53.477

According to above conclusions, the logistics company can be arranged based on the optimal route of delivery; goods can be transported by the shortest path to each demand point. The particle swarm algorithm is applied to logistics and distribution problems, with a relatively low cost to walk goods timely delivery to all demand points, can be realized.

C. Results of Analysis

In the 100 operations, the optimal path can up to 93 times, unknot are 7 times, the optimal path for 6.52s



Profile Summary

Generated 12-May-2013 17:21:11 using cpu time.

Function Name	Calls	Total Time	Self Time*	Total Time Plot (dark band = self time)
main	1	6.520 s	0.197 s	
fitness	4329	6.323 s	0.352 s	
f	12987	5.971 s	5.971 s	

Self time is the time spent in a function excluding the time spent in its child functions. Self time also includes overhead resulting from the process of profiling.

Fig 3: Running time

The experimental results show that, the success rate of PSO method in solving problems of searching is 93%. If you want to improve the searching success rate, only need to increase the value of MaxDT, but this will decrease the searching rate.

V. Conclusion

In this paper, we optimized the vehicle routing problem by improved particle swarm algorithm, and achieved good results which are consistent with reality. The experiment proved that it is more likely to success if use multi-swarm particle swarm to solve VRP problem .Our research contribute to the problem of large-scale reverse transport and logistics network optimization problem in high complexity transportation context, which helps to improve efficiency. Improved algorithm is better for VRP problem, and this manifested itself in the paper.

References

Li ning, Zou Tong Sun Debao etc. vehicle routing problem with time Windows of particle swarm algorithm [J]. Systems engineering theory and practice, 2004, 24 (4): 130-135.

Wu Bin, Ni Weihong Fan Shuhai etc. Open dynamic network vehicle routing problem of particle swarm algorithm [J]. Journal of computer integrated manufacturing system, 2009, (9): 1788-1794.

Ge Hongyi ZhenTong, Jiang Yuying etc. Grain logistics vehicle routing problem with time Windows research [J]. Computer engineering and applications, 2009, (35): 212-10.3778 DOI: 10.3778 / j.i SSN 1002-8331.2009.35.064.

Wei Ming JinWenzhou.To solves the vehicle routing problem of discrete particle swarm algorithm [J]. Journal of computer science, 2010, 5 (4): 187-10.3969 DOI: 10.3969 / j.i SSN. 1002-137-X.2010,04,047.

Wu Yaohua Zhang Nianzhi. Vehicle routing problem with time Windows of the improved particle swarm algorithm research [J]. Computer engineering and application, 2010 46-48 (15) : 230-10.3778 DOI: 10.3778 / j.i SSN 1002-8331.2010.15.068.

Wang Tiejun Wu KaiJun. Yard more vehicle routing problem of improved particle swarm algorithm [J]. Computer engineering and applications, 2013, (2). DOI: 10.3778 / j.i SSN. 1002-8331.1207-0355.

LiYi, Lu Beichuan, LiuChunXu etc. Vehicle routing problem of chaotic particle swarm optimization (pso) algorithm study [J]. Journal of chongqing jiaotong university (natural science edition), 2012, 31 (4) : 842-10.3969 DOI: 10.3969 / j.i SSN 1674-0696.2012.04. 26.

Su Mengluo Yang Hongan, Sun Qifeng etc. For solving the vehicle routing problem unmounted integration hybrid particle swarm algorithm [J]. China's manufacturing industry informatization, 2012, 9 (9) : 52-56. DOI: 10.3969 / jissn. 1672-1616.2012.09.15

Zhang Siliang GeHongWei. A hybrid approach to solve the vehicle routing problem [J]. Computer engineering and applications, 2012, 13 (8) : 226-10.3778 DOI: 10.3778 / j.i SSN 1002-8331.2012.08.064.

Wang Tiejun WuKaiJun. Yard more vehicle routing problem of improved particle swarm algorithm [J]. Computer engineering and applications, 2012,.

Wang Ya, Li Dan, etc. Improved chaotic particle swarm optimization (pso) algorithm to solve the vehicle routing problem [J]. Computer application research, 2011, 28 (11) : 4107-10.3969 DOI: 10.3969 / j.i SSN 1001-3695.2011.11.028.

Wang Kefeng,Ye Chunming, Li Yonglin etc. At the same time send pickup vehicle routing problem algorithm research review [J]. Computer application research, 2013, 30 (2) : 334-10.3969 DOI: 10.3969 / j.i SSN 1001-3695.2013.02.004.

XiaoLi BaoJunJie. An improved particle swarm algorithm in the application of the logistics distribution routing problem [J]. Journal of hunan university of science and technology (natural science edition), 2012, 27, (2) : 88-88. The DOL: 10.3969 / j.i SSN 1672-9102.2012.02.018.

Huang Hong. Ant algorithm based on particle swarm - the random demand of vehicle routing problem research

Fujian agriculture and Forestry University, 2011.

Wang Tiejun WuKaiJun. Yard more vehicles routing with time Windows of particle swarm optimization algorithm [J]. Computer engineering and applications, 2012,.

Wen Huiying Sun Bo. Collaborative vehicle routing problem based on discrete particle swarm algorithm [J]. Highway traffic science and technology, 2011, 28 (1) : 149-10.3969 DOI: 10.3969 / j.i SSN 1002-0268.2011.01.028.

Huang Zhen,Zen Kehan, GuZhiwen etc. A hybrid algorithm application in vehicle routing problem [J]. Journal of huizhou university, 2012, 32 (6) : 63-67.

KouMingShun Ye Chunming, Chen Zihao etc. Application of bee breeding evolutionary particle swarm optimization (pso) algorithm to solve the vehicle routing problem [J]. Journal of industrial engineering, 2012, 15 (1) : 23-27. DOI: 10.3969 / j.i SSN 1007-7375.2012.01.005.

Yang Lingyun. Improved particle swarm algorithm in the application of vehicle routing problem research [D]. Henan university, 2011.

GAZALANAAZ QURESHI,DR. P. R. BAJAJ,P. V. PURANIK.Particle Swarm Optimization with Genetic Operators for Vehicle Routing Problem[J].International Journal of Engineering Science and Technology,2012,4(7).

S. Geetha, G. Poonthalir and P. T. Vanathi, "A Hybrid Particle Swarm Optimization with Genetic Operators for Vehicle Routing Problem", Journal of Advances in Information Technology, vol. 1, no. 4, November 2010.

VRP Web http://neo.lcc.uma.es/radi-aeb/WebVRP/.