Particle Swarm Optimization
In Solving Capacitated Vehicle Routing Problem

M. M. Tavakoli
MSc. CSE Department of Shiraz University, Kerman, Iran
mtavakoli@cse.shirazu.ac.ir

Paper Reference Number: 07-50-6060
Name of the Presenter: Mohammad Mahdi Tavakoli

Abstract

The Capacitated Vehicle Routing Problem (CVRP) is a NP – Complete problem and according to this definition there is no exact solution for it. So researchers try to achieve a near optimum solution for it, by using meta – heuristic algorithms. The aim of CVRP is to find optimum route for every vehicle as a sequence of customers, that vehicle served. We employ Particle Swarm Optimization to solve this problem. In following of this paper we completely explain how we adjust PSO for a discrete space problem like CVRP, and the process of tweaking solutions, and at last for evaluation of our approach and show the effectiveness of our work, we show the result of running proposed approach over benchmarking data set of capacitated vehicle routing problem.

Key words: Capacitated Vehicle Routing Problem (CVRP), Particle Swarm Optimization (PSO), Traveling Salesman Problem (TSP), Meta – heuristic, Euclidean Distance

1. Introduction

The Vehicle Routing Problem (VRP) was first proposed by Dantzig and Ramser (1959) and has been widely studied. According to L. Guerra et al. (2007) and S. Masrom et al. (2010), VRP is a combinatorial optimization problem in which a set of routes for a fleet of delivery vehicles based at one or several depots must be determined for a number of customers. The main objective of VRP is to serve customer demands by a minimum cost vehicle routes originating and terminating in a depot. Several variations of the VRP exist in order to adapt to various practical characteristics and constraints such as: Multiple Depot VRP (MDVRP), Split Delivery VRP (SDVRP), Dynamic VRP (DVRP), Now if we have a constraint on capacity of every vehicle, the problem is known as capacitated vehicle routing problem (CVRP).

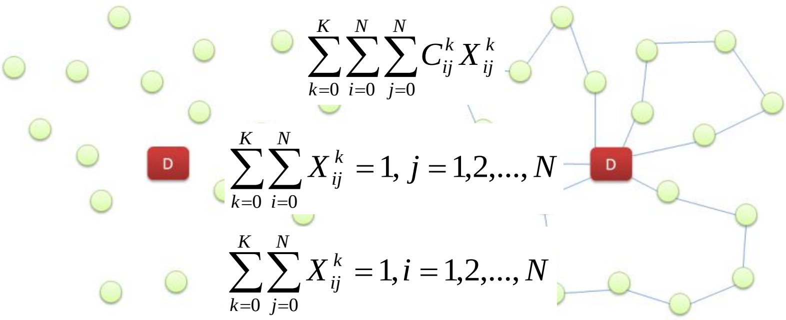


Fig. 1: CVRP with one depot and 18 customers

1-1. CVRP Model:

Capacitated Vehicle Routing Problem as defined by J. F. Cordeau (2002) and J. Lysgaard (2004) is a set of N customers with determined demands which must served from common depot by fleet of delivery vehicles that has constraint on their capacity. The cost of a specific vehicle V_i after completing a tour from depot and serving some customers in its route is the summation of Euclidean distance between each pair of nodes that vehicle visit.

The objective of CVRP is to find a collection of simple circuits in the graph of problem (each circuit corresponding to a vehicle route) with a minimum cost such that:

- Each customer is served exactly once, and by exactly one vehicle
- Each vehicle route leaves from and returns to depot
- Sum of the demands of the customers visited by each vehicle route does not exceed given vehicle capacity C.

Assume that depot is 0 and we should serve N customers by K vehicles. The demand of customer C_i is q_i, the capacity of vehicle k is Q_k and the maximum travel distance by vehicle k is D_k. The mathematical model of CVRP L. Bodin (1983) is described as follows:
If vehicle k travel from customer i to j, X_{ij}^{k} =1and else the value is 0.

Objective functic

$$\sum_{k=0}^{K}\sum_{i=0}^{N}\sum_{j=0}^{N} C_{ij}^{k} X_{ij}^{k} \tag{1}$$

$$\sum_{k=0}^{K}\sum_{i=0}^{N} X_{ij}^{k} = 1,\ j = 1,2,\dots,N \tag{2}$$

$$\sum_{k=0}^{K}\sum_{j=0}^{N} X_{ij}^{k} = 1, i = 1,2,\dots,N \tag{3}$$

$$\sum_{i=0}^{N} X_{it}^{k} - \sum_{j=0}^{N} X_{tj}^{k} = 0, k = 1,2,\dots,K; t = 1,2,\dots,N \tag{4}$$

$$\sum_{i=0}^{N}\sum_{j=0}^{N} d_{ij}^{k} X_{ij}^{k} \le D_k, k = 1,2,\dots,K \tag{5}$$

$$\sum_{j=0}^{N} q_j \left(\sum_{i=0}^{N} X_{ij}^{k} \right) \leq Q_k, k = 1,2,..., K \tag{6}$$

$$\sum_{j=1}^{N} X_{0j}^{k} \leq 1, k = 1,2,..., K \tag{7}$$

$$\sum_{i=1}^{N} X_{i0}^{k} \leq 1, k = 1,2,..., K \tag{8}$$

$$X_{ij}^{k} \in \{0,1\}; i, j = 0,1,..., N; k = 1,2,..., K \tag{9}$$

The number of customers determined by N, number of vehicles is K, and the travelling cost by vehicle k from customer i to j is show as C_{ij}^{k}, and d_{ij}^{k} is the travel distance between customer i to j.

The objective function Eq. (1) is to minimize the total cost by all vehicles that is sum of the travel distance of each vehicle, and the travel distance for each vehicle is sum of Euclidean distance between each pair of customers. Constraints Eq. (2), Eq. (3) ensure that each customer is served exactly once. Constraint Eq. (4) ensures the connectivity of the route. Constraint Eq. (5) shows that the total length of each route has a limit. Constraint Eq. (6) shows that the total demand of any route must not exceed the capacity of the vehicle. Constraint Eq. (7) and Eq. (8) ensure that each vehicle is used no more than once. Constraint Eq. (9) ensures that the variable only takes the integer 0 or 1.

1-2. Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a global optimization technique. It is originally attributed to Kennedy and Eberhart (1995). A swarm consist of a set of particles that each particle represents a potential solution. Assumes that each solution can be represented as a point in N – Dimensional space that each point or particle has an initial velocity, Particles move through solution space, after each time step, particles are evaluated according to some fitness criterion. They are accelerated towards particles with better fitness values within their communication group. This property of PSO help particles escape from local optimal solutions. Each particle has a simple memory that remember the position of best solution achieve by itself, this value is called personal best (pbest) and the position of best solution obtained so far by any particle in the neighborhood of that particle, that known as global best (gbest). The basic concept of PSO lies in accelerating each particle toward its pbest and the gbest locations, with a random weighted acceleration at each time step.

2. Research Methodology

In this section the main assumptions for solving CVRP by PSO are given and in following of section our methodology is explained in detail.

2-1. Assumptions

We assume that width and height dimensions of our problem space initialize to 100.

Different neighborhood types have been defined for Particle Swarm Optimization such as Global, Geographical and Social neighborhood, that we choose Global neighborhood as our neighborhood selection for solving CVRP.

The random weighted acceleration values for pbest (W_1) and gbest (W_2) are initializing by generating random numbers multiply by constant values C_1 and C_2 respectively that

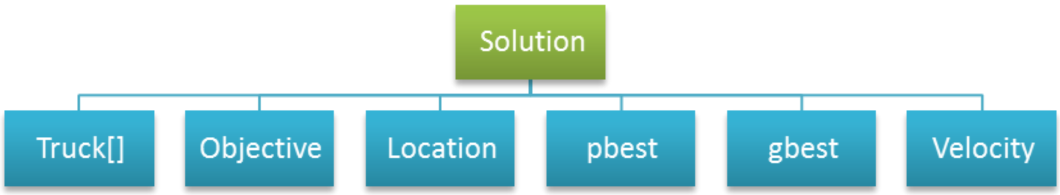
usually set them in a way which summation of them is 4. Z. Ying et al. (2003) and S. R. Venkatesan (2011) set C_1 and C_2 to 2, but we empirically set C_1 to 1.5 and C_2 to 2.5.
Weight for pbest: $W_1 = C_1$.Rand ()
Weight for gbest: $W_2 = C_2$.Rand ()
The inertia weight has a well balance mechanism with flexibility to enhance and adapt to both global and local exploration. Large inertia weight facilitates global exploration and small value of it, enhance local exploration. We set the inertia to 0.47.

2-2. Methodology

The steps of our revised PSO algorithm for solving CVRP are listed below:

1. Initialize Particles
    a. Generating a set of solutions in a random greedy manner and assign a particle to each of them.
    b. Evaluate the fitness of each particle
    c. Randomly locating each particle in problem space
2. Adjust positions
    a. Each particle tries to modify its position using the following information:
        - the current positions,
        - the current velocities,
        - the distance between the current position and pbest,
        - the distance between the current position and the gbest
    b. We compute new position components of particle P_i (x-axis, y-axis) by using equations below:
        - new X = inertia * P_i.Velocity + [(W_1 * (P_i.pbest.X – P_i.X)) + (W_2 * (P_i.gbest.X – P_i.X))] + P_i.X
        - new Y = inertia * P_i.Velocity + [(W_1 * (P_i.pbest.Y – P_i.Y)) + (W_2 * (P_i.gbest.Y – P_i.Y))] + P_i.Y
3. Find nearest neighbor of each particle to this new calculated position and tweak solution of it by using TOE, TOI and TSPOE heuristics that we completely explain them in next section. If this tweak cause to improvement in best fitness obtained so far by algorithm, current particle sets its solution to its nearest neighbor's tweaked solution and the particle update its location to point (new X, new Y).
4. Evaluate fitness of each particle by calculating the summation of each vehicle cost.
5. Check if new pbest or gbest values achieved and update them.
6. Go to step 2 and repeat this process until the current iteration violate the max iteration constraint.

2-3. Model CVRP and solution encoding



We write our program in an object oriented manner. In object oriented programming each object in actual world represented as a class that properties and functionalities of that object are modeled as fields and methods respectively.

We assume each vehicle (or truck in our source code) has a capacity and route. With this definition each solution is constitute of vehicles (trucks) that the objective of solution is summation of each vehicle travel distance, and other attributes listed in fig. 2 are something needed for a particle.

Fig. 2: Solution encoding for CVRP by PSO

In fig. 3 you see an example of sol his benchmark there are 32 customers that should be served by 5 vehicles from one depot. We assume that the number of depot is one, and to show each vehicle start from depot and return to it, we put 1 in beginning and end of each vehicle route. Also the remaining capacity of each vehicle and travel distance of it, is shown in solution form.

Truck ID	Route	Capacity	Distance
1	1 - 28 - 25 - 1	56	59.26263
2	1 - 31 - 17 - 8 - 2 - 13 - 1	12	88.50539
3	1 - 21 - 6 - 26 - 11 - 16 - 10 - 23 - 30 - 1	9	236.7426
4	1 - 7 - 18 - 20 - 32 - 22 - 14 - 27 - 1	6	180.9489
5	1 - 15 - 19 - 9 - 12 - 5 - 29 - 24 - 3 - 4 - 1	7	231.9918

Fig. 3: An example of solution for A-n32k5

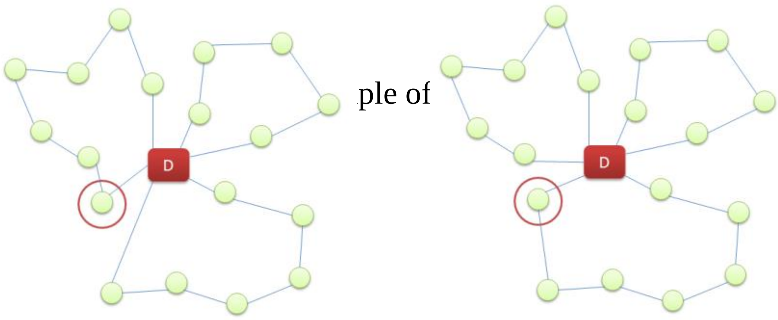
2-4. Process of tweaking solutions

We use three heuristics to improve solutions. A brief description of them is mentioned below:

- Two Optimal Exchange (TOE): first by generating two random numbers, two different vehicles selected and again by generating two random numbers we select one customer in route of each selected vehicles. Then if by exchanging these two customers no violation occurs in capacity of vehicles (in other word the result be feasible) the result return as new solution, else we repeat this process to find a feasible combination.

ple of

- Two



Optimal Insertion (TOI): similar to TOE, first by generating two random numbers two different vehicles selected and again by generating one random number we select one customer in route of first vehicle. If no violation occurs in capacity of second vehicle, we remove selected customer from first vehicle route and insert it beside of nearest neighbor of that customer in second vehicle route. Here we must pay some small cost to find nearest neighbor of selected customer in second vehicle route, but it cause improvement in quality of generated solution. Like TOE heuristic this process repeat until we find a feasible combination.

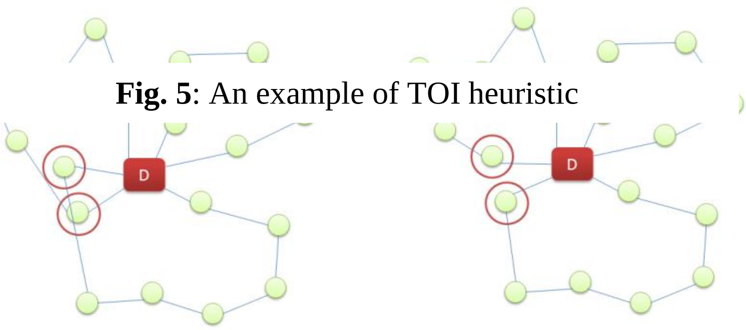


Fig. 5: An example of TOI heuristic

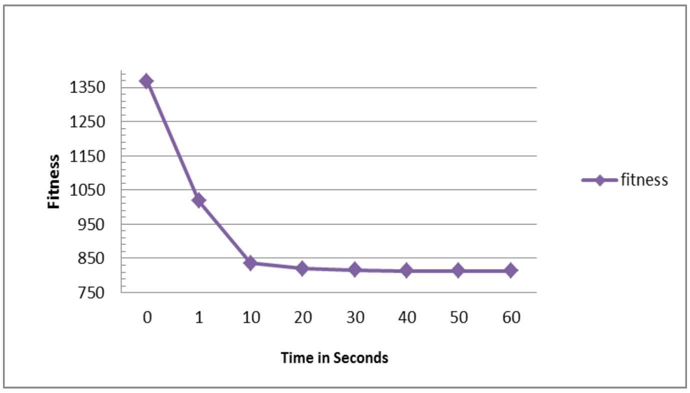
- TSP

Optimal Exchange (TSPOE): this heuristic applied to all vehicles in problem instance. This is a contribution in process of tweaking solutions that in each step we assume route of each vehicle as a TSP problem and try to improve quality of solution by exchanging customers in route of each vehicle. In other word, we try to find pair of customers iteratively that exchanges of them lead to improvement in quality of solution.

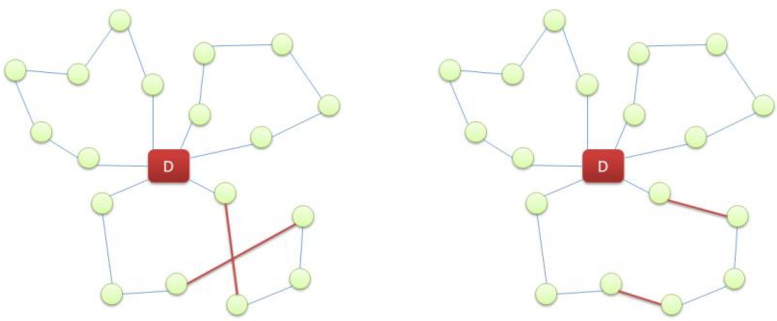
Fig. 7: Evolution of solutions in A-n80k10

Fig. 6: An example of TSPOE heuristic

3. Results and Analysis



We write our algorithm with Microsoft Visual Studio 2008 and by using C# language of this IDE. In figs. 7 and 8 we show the average process of running proposed approach over A-n80k10 and A-n32k5 benchmarks respectively, for ten runs and less than one minute.



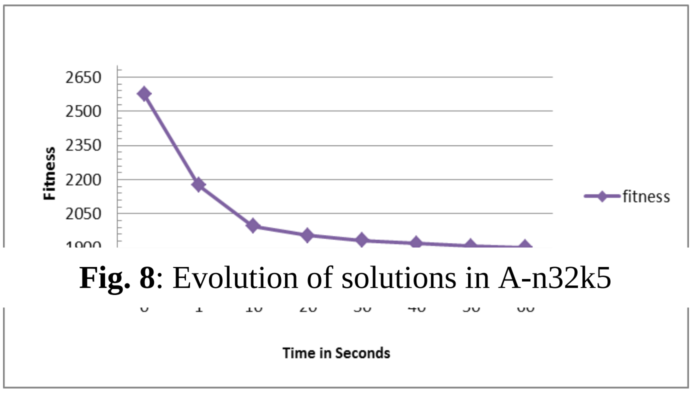


Fig. 8: Evolution of solutions in A-n32k5

In table 1 we show the best results obtained by proposed approach over 7 benchmarks of CVRP. We compare results of our approach with best known solution (BKS) of each benchmark.

Benchmark	# Customers	# Vehicles	BKS	Proposed approach
P-n16k8	16	8	450	451.34
P-n20k2	20	2	216	217.42
A-n32k5	32	5	784	787.08
A-n44k6	44	6	937	938.17
A-n61k9	61	9	1034	1050.38
A-n80k10	80	10	1766	1795.09
F-n135k7	135	7	1166	1241.29

Table 1. Best results of proposed approach

4. Conclusions

In this paper we presented a revised PSO algorithm for solving CVRP. We use combination of different heuristics such as TOE, TOI and TSPOE, that as you see in table 1, they lead to greatly enhanced solutions and obtain good results in both small and medium problem instances. But in case of large problems like F-n135k7, it seems that the algorithm trapped in local optima and could not find near optimum solutions. The future work could be an improvement of proposed approach in a way that performs well on large benchmarks too.

References

Dantzig, G.B., Ramser, J.H. (1959). The Truck Dispatching Problem Management Science 6 (1): 80–91. doi:10.1287/mnsc.6.1.80. ISSN 0025-1909. JSTOR 2627477

Guerra, L., Murino, T., & Romano, E. (2007). A heuristic algorithm for the constrained location - routing problem, International Journal of Systems Applications, Engineering & Development, Vol. 1, No. 4, pp. 146-154

Masrom, S., Abidin, Z. Z., Hashimah, P. N., & Rahman, A. S. (2010). Providing flexibility in the Vehicle Route Optimization for the AVL based Transport Monitoring System, WSEAS Transactions on Computers, Vol. 9, pp. 526-535

Cordeau, J. F., Gendreau, M., Laporte, G., Potvin, J. Y., & Semet, F. (2002). A guide to vehicle routing heuristics. J. Operat. Res. Soci., 53: 512-522. DOI: 10.1057/ palgrave.jors.2601319

Lysgaard, J., Letchford, A. N., & Eglese, R.W. (2004). A new branch-and-cut algorithm for the capacitated vehicle routing problem. Mathem. Programm., 100: 423-445. DOI: 10.1007/s10107-003-0481-8

Bodin, L., Golden, B., Assad, A., & Ball, M. (1983). Routing and scheduling of vehicles and crews: The state of the art, Computers and Operations Research, Vol. 10, No. 2, pp. 63-211

Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization Proceeding of IEEE International Conference on Neural Network, (ICNN'95), Perth, Australia, pp: 1942-1948

Ying, z., guangjie, z., & feihong Yu. (2003). particle swarm optimization-based approach for optical finite impulse response filter design, applied optics issn 0003-6935 coden apopai, vol. 42, no8, pp. 1503-1507

Venkatesan, S. R., Logendran, D., & Chandramohan, D. (2011). Optimization of capacitated Vehicle Routing Problem using PSO, International Journal of Engineering Science and Technology (IJEST), Vol. 3 No.10