A HYBRID PSO ALGORITHM FOR THE CVRP PROBLEM

Yucheng Kao and Mei Chen
Department of Information Management, Tatung University, 104, Taipei, Taiwan

Keywords: Vehicle routing problems, Particle swarm optimization, Simulated annealing.

Abstract: The Capacitated Vehicle Routing Problem (CVRP) has been studied over five decades. The goal of CVRP is to minimize the total distance travelled by vehicles under the constraints of vehicles' capacity. Because CVRP is a kind of NP-hard problem, a number of meta-heuristics have been proposed to solve the problem. The objective of this paper is to propose a hybrid algorithm combining Combinatorial Particle Swarm Optimization (CPSO) with Simulated Annealing (SA) for solving CVRP. The experimental results show that the proposed algorithm can be viewed as an effective approach for solving the CVRP.

1 INTRODUCTION

There are many combinatorial Optimization problems in the real world, including portfolio optimization, job dispatching, vehicle routing problems, etc. The vehicle routing problem (VRP) is an interesting research subject. Many researchers have developed a variety of solution approaches to solve the VRP over past 50 years. The objective of capacitated vehicle routing problem (CVRP) is to minimize the total distance traveled by vehicles. There are two stages used in common CVRP approaches: customer clustering and sequencing. In clustering stage we assign each customer into a vehicle, while in sequencing stage we arrange the visiting order for a vehicle which has been assigned customers. Apparently, customer clustering results affect sequencing results, and the sequencing results determine the objective function values.

CVRP belongs to the category of NP-hard problems. In recent decades meta-heuristic approaches attract many researchers' attention and have been applied to the VRPs. Meta-heuristic approaches include genetic algorithms, simulated annealing, ant colony optimization, particle swarm optimization, etc. One of important advantages of using these approaches is that we can obtain optimal or near optimal solutions in a reasonable computation time. Applying a meta-heuristic approach to solve the CVRP, we need to concern following aspects: current routing solutions may affect clustering results of the next iteration; initial solutions may affect the evolution result; checking solution feasibility may be necessary from time to time, solution quality and the converging speed of evolutionary processes are important performance criteria but both often conflict with each other.

Particle Swarm Optimization (PSO) was proposed by Kennedy and Eberhart in 1995. PSO has been applied to solve many problems, including the VRP. Chen et al. (2006) first proposed a discrete PSO (DPSO) approach to solve the CVRP. Their approach follows the two-stage approach, clustering first and routing second. They use DPSO to perform customer clustering and use SA to determine the visiting order of each vehicle. Due to long solution strings (the product of customer number and vehicle number) their algorithm often needs a larger amount of CPU time to find optimal solutions. Ai et al. (2009) also proposed an algorithm based on PSO for CVRP. The solution string of a particle represents coordinate points and coverage radius of vehicles. Vehicle routes are constructed based on these points and radiuses. The order of visiting customers of each route is found by using an insertion heuristic. Thus these two papers used the cluster-first-route-second approach to solve CVRP.

This paper proposes a new hybrid PSO with SA to solve the CVRP. Similarly, it follows the cluster-first-route-second approach: use a new discrete PSO to find customer clustering results and then use simulated annealing to arrange customer visiting orders. Experimental results show that the proposed algorithm can solve the CVRPs efficiently.

Kao Y. and Chen M..
A HYBRID PSO ALGORITHM FOR THE CVRP PROBLEM.
DOI: 10.5220/0003724005390543
In *Proceedings of the International Conference on Evolutionary Computation Theory and Applications* (FEC-2011), pages 539-543
ISBN: 978-989-8425-83-6

2 LITERATURE

2.1 VRP Problems

VRP is an interesting subject which belongs to a category of combinatorial optimization problems. Dantzig (1959) first proposed the truck dispatching problems. The VRP has many variants. Jozefowiez et al. (2008) presented detailed classifications and comparisons for VRP problems with respect to problem definitions, objectives and algorithms. The article mentioned that VRP has widely applied to transport delivery routing, urban school bus route planning, rural school bus routing planning, urban trash collection, etc. The authors introduced objectives of VRP from single objective to multiple objectives. One of common objective functions is to minimize the total distance traveled by all vehicles. Other objectives include workload balance, total number of vehicle, total traveling time and total waiting time of customers.

Classical VRP problems are classified into two categories: capacitated vehicle routing problems (CVRP) and vehicle routing problem with time windows (VRPTW). CVRP's objective is to find a set of routes with minimum total distance traveled to deliver goods for all customers with different demands. Its Constraints include: every customer is to be served exactly once by one vehicle; each vehicle starts and ends at the depot, every vehicle has a limited capacity.

VRP is a kind of NP-hard problems. Researchers developed many solution approaches to solve VRPs, such as exact algorithms and heuristic algorithms (Cordeau et al., 2007). Meta-heuristics are a kind of heuristics, which have widely been applied to VRPs. The performance of meta-heuristics is often better than classical heuristics. Popular meta-heuristics include Genetic Algorithm (GA), Tabu Search (TS), Simulated Annealing (SA), Ant colony systems (ACS), Particle Swarm Optimization (PSO), Scatter Search (SS), etc.

Baker et al. (2003) proposed a simple GA for solving CVRP. The length of solution string is equal to the total number of customers. Each gene has an integer number which ranges from 1 to the total number of vehicles. The algorithm selects parent solutions by using a binary tournament method, produces offspring solutions by using two-point crossover operation, mutates solutions by swapping two randomly selected genes, selects better solutions for next generation by using a ranking replacement method.

Bell et al. (2004) proposed an ACO algorithm for solving the CVRP. They proposed two methods to improve the performance of their algorithm, including local exchange and candidate list. The authors also proposed multiple ant colonies method for solving large size problems (more than 100 customers). Zhang et al. (2009) proposed an algorithm which integrates scatter search with ACO for solving CVRP. That paper uses scatter search as the framework and applies ACO to construct route solutions.

2.2 Particle Swarm Optimization (PSO)

PSO is a population-based evolution algorithm and is a member of swarm intelligence techniques. PSO was proposed by Kennedy and Eberhart (1995) and inspired by the birds' foraging behavior. The searching process is affected by current positions, personal best positions (pbest) and the best position of the flock (gbest). Original PSO algorithm has the ability of fast convergence whereas it does not guarantee to find the optimal solution eventually. After some modifications, one of popular PSO version proposed by Shi and Eberhart (1999) is defined as follows:

$$V_{id} = W \times V_{id} + C_1 \times Rand \times (P_{best} - X_{id}) + C_2 \times Rand \times (G_{best} - X_{id}) \quad (1)$$

$$X_{id} = X_{id} + V_{id} \quad (2)$$

This paper adopts a discrete PSO model, combinatorial particle swarm optimization (CPSO) to solve the CVRP problem. CPSO was proposed by Jarboui et al. (2007) for solving combinatorial optimization problems. It permits solutions transited from a discrete variable space to a continuous variable space. After a standard PSO evolution process, new continuous solutions are transited back to the discrete space in order to obtain new discrete solutions. It has been proved that CPSO is a useful solution approach for solving most of discrete combinatorial optimization problems.

3 CPSO-SA FOR CVRP

This paper proposes a new algorithm combining CPSO and SA, which is called CPSO-SA. It follows the two-stage approach for solving CVRP. CPSO deals with customer clustering while SA arranges customer visiting sequence. At the end of iterations, the algorithm conducts local searches for top three best particles. Particle's best solutions and the global best solution obtained at current iteration will be used to generate new particles in the next iteration.

This paper adopts similar basic approach used by Chen et al. (2006), but it tries to improve the latter by using a short solution representation and a more efficient discrete PSO algorithm. For customer clustering, it doesn't need to check solution feasibility to prevent that a customer is assigned to more than one vehicle, and, as a result, it does save some computational time.

3.1 Mathematical Model

The CVRP considered in this paper has a symmetric network. The objective is to minimize the total distance traveled by all vehicles. This problem has been formulated as follows:

Notation

0：depot；

n：total number of customers；

N：customers set，$N=\{1,2,...,n\}$；

v：total number of vehicles；

V：vehicles set，$V=\{1,2,...,v\}$；

d_{ij}：distance between customer I and j, $d_{ij}=d_{ji}$，$\forall i, j \in N \cup \{0\}$；

q_i：demand for customer I, $q_0 = 0$；

Q：maximum capacity for each vehicle；

X_{ij}^k：vehicle k travels edge i-j or not, 0: no, 1: yes；

R_k：customer set served by vehicle k，$R_k=\{j_{k,1}, j_{k,2}, ..., j_{k,|R_k|}\}$；

$|R_k|$: cardinality of R_k;

$j_{k,m}$：mth customer served by vehicle k.

Objective function

$$\text{Minimize} \sum_{k=1}^{v}\sum_{i=0}^{n}\sum_{j=0}^{n} d_{ij} X_{ij}^k \tag{3}$$

Subject to

$$\sum_{k=1}^{v}\sum_{i=0}^{n} X_{ij}^k = 1, \; j = 1, 2, ..., n \tag{4}$$

$$\sum_{k=1}^{v}\sum_{j=0}^{n} X_{ij}^k = 1, \; i = 1, 2, ..., n \tag{5}$$

$$\sum_{i=0}^{n} X_{iu}^k - \sum_{j=0}^{n} X_{uj}^k = 0, \; k = 1, 2, ..., v \,;\; u = 1, 2, ..., n \tag{6}$$

$$\sum_{j=1}^{n}\sum_{i=0}^{n} (q_j \times X_{ij}^k) \le Q \,,\; k = 1, 2, ..., v \tag{7}$$

$$\sum_{j=1}^{n} X_{0j}^k \le 1, \; k = 1, 2, ..., v \tag{8}$$

$$\sum_{i=1}^{n} X_{i0}^k \le 1, \; k = 1, 2, ..., v \tag{9}$$

Eqs. (4) and (5) ensure that each customer must be served by a vehicle exactly once. Eq. (6) considers that the continuity for every vehicle can be maintained. Eq. (7) ensures that the total customer demand of a vehicle does not exceed its maximum capacity. Eqs. (8) and (9) mean that every vehicle can be used once or not be used.

3.2 Solution Representation

This paper adopts a cluster-first-route-second approach for solving CVRP. Thus a solution has two sections. The first one is for customer clustering and the second one is for customer sequencing. The length of each of these two sections equals to n. $XV_p = (xv_{p1}, xv_{p2}, ..., xv_{pn})$ stands for the result of customer clustering of particle p; xv_{pd} indicates the vehicle which serves customer d. $Tour_p = (0, j_{1,1}, …, j_{1,|R1|}, 0, j_{2,1}, …, 0, j_{v,1}, …, j_{v,|Rv|}, 0)$ is the customer sequencing result of all vehicles found by particle p.

In the first stage, some other vectors have to be defined for executing PSO algorithms. $V_p = (v_{p1}, v_{p2}, ..., v_{pn})$ indicates the velocity of particle p; $P_p = (P_{p1}, P_{p2}, ..., P_{pn})$ indicates the personal best solution ever found of particle p; $G = (G_1, G_2, ..., G_n)$ indicates the global best solution ever found of the swarm.

3.3 CPSO-SA Algorithm

The proposed algorithm starts conducting evolution after initialization until the iteration number reaches the max iteration number. The latest global best solution is the final solution. Evolution includes the processes of customers clustering and sequencing. Customer clustering is conducted by using CPSO while the visiting order is conducted by using SA. Then the algorithm computes objective function value for every particle according to Eq. (3). After that, we perform a local search on top three best particles for solution improvement. The procedure is to choose two routes randomly with considering capacity constraints, to select a customer from each of selected routes, and then to exchange their vehicle numbers. After that, find new visiting orders of these two routes by using SA. If new solution is better, replace the current solutions with new solutions. Before the iteration ends, particles update their Pbest and Gbest solutions.

3.3.1 Customer Clustering

In every iteration, customer clustering solution is found by using Eqs.(10)~(16). *pop* particles with solutions *XV*s flying in a space of *n* dimensions take place between two transition phases: from discrete space to continuous space and from continuous space back to discrete space.

Transition Phase 1:

$$y_{pd}^{iter} = \begin{cases} 1 & \text{if}\,(xv_{pd}^{iter-1} = G_d^{iter-1}) \\ -1 & \text{if}\,(xv_{pd}^{iter-1} = P_{pd}^{iter-1}) \\ -1 \text{ or } 1 \text{ randomly} & \text{if}\,(xv_{pd}^{iter-1} = P_{pd}^{iter-1} = G_d^{iter-1}) \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

Flying Phase:

$$d_1 = -1 - y_{pd}^{iter}, \text{ distance between } xv_{pd}^{iter-1} \text{ and } P_{pd}^{iter-1} \quad (11)$$

$$d_2 = 1 - y_{pd}^{iter}, \text{ distance between } xv_{pd}^{iter-1} \text{ and } G_d^{iter-1} \quad (12)$$

$$v_{pd}^{iter} = w \cdot v_{pd}^{iter-1} + c1 \cdot r1 \cdot d1 + c2 \cdot r2 \cdot d2 \quad (13)$$

$$\lambda_{pd}^{iter} = y_{pd}^{iter} + v_{pd}^{iter} \quad (14)$$

Transition Phase 2:

$$y_{pd}^{iter} = \begin{cases} 1 & \text{if}\,(\lambda_{pd}^{iter} > \alpha) \\ -1 & \text{if}\,(\lambda_{pd}^{iter} < -\alpha) \\ 0 & \text{otherwise} \end{cases} \quad (15)$$

$$xv_{pd}^{iter} = \begin{cases} G_d^{iter-1} & \text{if}\,(y_{pd}^{iter} = 1) \\ P_{pd}^{iter-1} & \text{if}\,(y_{pd}^{iter} = -1) \\ \text{any vehicle} & \text{otherwise} \end{cases} \quad (16)$$

Where, *iter* is the current number of iteration；*p* and *d* represent particle number and dimension number; y_{pd}^{iter} is a dummy variable for dimension *d* of particle *p* in iteration *iter*； G_d^{iter} is the value of dimension *d* for the gbest solution in iteration *iter*； P_{pd}^{iter} is the value of dimension *d* of the pbest solution for particle *p* in iteration *iter*； xv_{pd}^{iter} ：is the value of dimension *d* of particle *p* in iteration *iter*；*c1* and *c2* are weighting coefficients； v_{pd}^{iter} is the velocity for dimension *d* of particle *p* in iteration *iter*；*r1* and *r2* are random numbers between 0 and 1； λ_{pd}^{iter} ：value of dimension *d* for particle *p* in iteration *iter*； α is a threshold parameter.

3.3.2 Customer Sequencing

Because capacity constraints have been considered in the first stage, this stage just needs to consider the total distance traveled by vehicles. Thus the problem of customer sequencing for a vehicle is equivalent to a TSP problem. Initial solutions are generated by using a greedy method and then SA algorithm, Eqs. (17) and (18), is used to improve initial solutions. *P*(*S'*) is the probability that SA accepts new solution *S'* which is worse than the current solution *S*.

$$\Delta = f(S') - f(S) \quad (17)$$

$$p(S') = \exp(-\Delta / t) \quad (18)$$

4 EXPERIMENTS

To verify our approch, we compare our algorithm with that proposed by Chen et al. in terms of solution quality and CPU time. CPSO-SA was coded in Java and executed on a PC with 3.5GB of RAM and Intel Core 2 CPU E8400 3GHz. The CPSO-SA parameters used for all CVRP problems are pop = the amount of customers, w = 0.8, C_1 = 1.1 , C_2 = 1.4, max iteration number = 300, $t_0 = 3$, t_f = 0.01, L = $n \times 2$, $\theta = 0.8$. Five CVRP test problems are collected from the website (http://www.branchand cut.org/VRP/data/).

Figure 1 presents the convergence trend of running CPSO-SA for solving the first test problem. The results of five test problems is listed in Table 1. To compare computational times, Chen's CPU times have been properly converted based on an equation (Patterson and Hennessy, 2005). The results show that CPSO-SA's solutions are equal to those obtained by Chen's approach with much less CPU times.

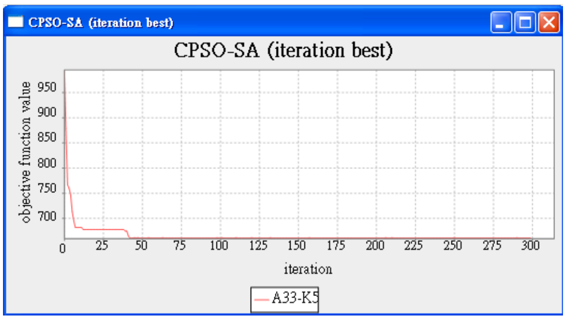

Figure 1: Convergence trend when CPSO-SA solving problem A33-K5.

Table 1: Experimental results.

No.	Problem	Cus. #	Vehicle #	Objective function value			CPU time (seconds)	
				BKS	DPSO-SA	CPSO-SA	DPSO-SA	CPSO-SA
1	A-n33-k5	32	5	661	661	661	19.4	**0.7**
2	A-n46-k7	45	7	914	914	917	77.3	**2.4**
3	A-n60-k9	59	9	1354	1354	1354	185.3	**6.5**
4	B-n35-k5	34	5	955	955	955	22.6	**1.2**
5	B-n45-k5	44	5	751	751	751	80.5	**4.8**

BKS: the best known solution in the literature
DPSO-SA used Intel Pentium IV CPU 1.8 GHz with 256M RAM
CPSO-SA uses Intel Core 2 CPU E8400 3GHz with 3.5G RAM

5 CONCLUSIONS

In our proposed approach, CPSO is first used to cluster customers into different vehicles and then SA is used to arrange the visiting order of each vehicle. A new solution representation is also proposed in order to reduce computing time which is necessary for judging feasibility and repairing infeasible solutions. Experimental results show that the CSPO-SA algorithm can effectively solve the CVRP problem within a more reasonable time period.

REFERENCES

Ai, T. J., Kachitvichyanukul, V., 2009. Particle swarm optimization and two solution representations for solving the capacitated vehicle routing problem. *Computers & Industrial Engineering (56)*, pp. 380-387.

Baker, B. M. and Ayechew, M. A., 2003. A genetic algorithm for the vehicle routing problem. Computers & Operations Research (30), pp. 787-800.

Bell, J. E. and McMullen, P. R., 2004. Ant Colony Optimization techinques for the vehicle routing problem. *Advanced Engineering Informatics (18)*, pp. 41-48.

Chen, A. L., Yang, G. K. and Wu, Z. M., 2006. Hybrid discrete particle swarm optimization algorithm for capacitated vehicle routing problem. *Journal of Zhejiang University SCIENCE A* 7(4), pp. 607-614.

Cordeau, J. F., Laporte, G., Savelsbergh, M. W. P. and Vigo, D., 2007. Chapter 6 Vehicle Routing. In: Barnhart, C. and Laporte, G. eds. *Handbook in Operations Research and Management Science*(14), pp. 367-428.

Jarboui, B., Cheikh, M., Siarry, P. and Rebai, A., 2007. Combinatorial particle swarm optimization (CPSO) for partitional clustering problem. *Applied Mathematics and Computation (92)*, pp. 337-345.

Jozefowiez, N., Semet, F. and Talbi, E. G., 2008. Multi-objective vehicle routing problems. *European Journal of Operational Research* (189), pp. 293-309.

Kennedy, J. and Eberhart, R., 1995. Particle swarm optimization. *Proceedings of IEEE International Conference on Neural Networks*, pp. 1942–1948.

Marinakis, Y., Magdalene, M. and Dounias, G., 2010. A hybrid particle swarm optimization algorithm for the vehicle routing problem. *Engineering Applications of Artificial Intelligence* (23:4), pp. 463-472.

Osman, I. H., 1993. Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem. *Annals of Operations Research*, pp. 421-451.

Patterson D. A., Hennessy J. L. (Eds.), 2009. Computer organization and design: the hardware/software interface. *Burlington, MA: Morgan Kaufmann Publishers*.

Solomon, M. M., 1987. Algorithms for the Vehicle Routing and Scheduling Problems with Time Window constraints. *Operations Research (35:2)*, pp. 254-265.

Zhang, X. and Tang, L., 2009. A new hybrid ant colony optimization algorithm for the vehicle routing problem. *Pattern Recognition Letters (30)*, pp. 848-855.