

A Novel PSO For Solving The VRPTW With Real Case Study

Shahrzad Amini

Department of Industrial Engineering

Islamic Azad University - South Tehran Branch, Tehran, Iran

Abstract

During the past few years, there have been tremendous efforts on improving the cost of logistics using varieties of models for vehicle routing problems. In fact, the recent rise on fuel prices has motivated many to reduce the cost of transportation associated with their business through an improved implementation of VRP systems. In this paper, we study the VRP with time windows. We propose a particle swarm optimization algorithm to solve the given VRPTW. A computational experiment is carried out by running the proposed PSO with the VRPTW benchmark data set of Solomon. The associated results show that this algorithm is able to provide good solutions that are very close to its optimal solutions for problems with 25 customers within reasonably computational time. Furthermore, our proposed PSO is used for a real-world case study of a Chlorine Capsule distribution company to the water reservoir in Tehran. The related results indicate that the algorithm can reduce the cost and time significantly.

Keywords

Particle Swarm Optimization, Vehicle Routing Problem with Time Windows, Real Case Study

1. Introduction

The vehicle routing problem was proposed for the first time by Dantzig and Ramser [1] in 1959. This problem is to find the optimal routes of delivery or collection from one or several depots to a number of cities or customers while satisfying some constraints. Collection of household waste, gasoline delivery trucks, goods distribution, snow plough and mail delivery are the most used applications of VRPs. The VRP plays a vital role in distribution and logistics. The interest in this problem is motivated by its practical relevance as well as by its considerable difficulty. The VRPTW is an extension of the classical VRP, encountered very frequently in making decision about the distribution of goods and services. This problem involves a fleet of vehicles set-off from a depot to serve a number of customers, at diverse geographic locations, with various demands and within specific time windows before returning to the depot eventually. Theoretically, this is an NP-hard problem, which means that it is believed that one may never find a computational technique that guarantees an optimal solution to larger instances for such problems. Because all kinds of exact algorithms are not efficient for solving NP-hard problems, several heuristics (or meta-heuristics) have been proposed for the VRP. A number of meta-heuristics, such as simulated annealing, genetic algorithm, tabu search, ant colony optimization and particle swarm optimization are designed for VRPs. PSO is a meta-heuristic algorithm that simulates the flight of flocks of birds, making good effect for the various multi-dimensional continuous space optimization problems, but less in research and application of discrete field.

2. Literature Review

The VRPTW is a generalization of the well-known VRP. Ellabib et al. [2] reviewed it as a combined vehicle routing and scheduling problem that often arises in many real-world applications. It is to optimize the use of a fleet of vehicles that must make a number of stops to serve a set of customers, and to specify which customers should be served by each vehicle and in what order to minimize the cost, subject to vehicle capacity and service time restrictions. The VRPTW has been the subject of intensive research efforts for both heuristic and exact optimization approaches. Early surveys of solution techniques for the VRPTW can be found in Golden and Assad [3], Solomon [4] and Chiang and Russell [5]. Because of the high complexity level of the VRPTW and its wide applicability to real-life situations, solution techniques capable of producing high-quality solutions in limited time, i.e. heuristics are of prime importance. Taillard et al. [6] have used heuristic for the VRP without time windows. In Taillard et al. [6] reported different solutions to the classical VRP by TS. A recent study in Compbell and Savelsbergh [7] has reported about insertion heuristics that can efficiently handle different types of constraints including time windows and multiple uses of vehicles.

2.1. Particle swarm optimization

PSO is a population-based search method proposed by Kennedy and Eberhart [8] in 1995. The main motivation came from the behaviour of group organisms, such as bee swarm, fish school, and bird flock. In PSO, a problem solution is represented by the position value of a multi-dimensional particle. The particle's velocity represents particle searching ability. Comprehensive details of the PSO mechanism and applications are provided by Kennedy and Eberhart [9] and Clerc [10]. As an emerging population-based search method, PSO has been recently applied to many operational research problems, such as flow shop scheduling, job shop scheduling, home care worker scheduling, multi-mode resource-constrained project scheduling, and vehicle routing. Some of the applications are briefly described below. Chen et al. [11] applied PSO for solving the CVRP. They applied PSO with the discrete value of particle's position and hybridized it with SA. Qing et al. [12] applied PSO for the VRPTW; however, the effectiveness of PSO on this problem has not been well proven since it is only demonstrated for solving a very small problem case. Tasgetiren et al. [13] applied PSO to solve the flow shop scheduling problem. They used a heuristic called the Smallest Position Value (SPV) to transform position value of particle to a job sequence. Alvarenga et al. [14] proposed a two-stage heuristic approach for solving the VRPTW. Their approach was involving an efficient GA and a set partitioning formulation. Their computational results outperformed the existing heuristic methods in terms of the minimal travel distance.

3. PSO framework for solving the VRPTW

The PSO algorithm starts by initializing a swarm of particles followed by a series of iteration steps to move the particles towards the best position. It is noted that it works on finding the best position of particles. So, to apply PSO for a specific problem, the relationship between the position particles and the problem solution must be clearly defined. A method for converting particle position to the problem solution is usually called 'decoding method'. The PSO framework for solving VRPTW is proposed based on the GLNPSO, a PSO Algorithm with multiple social learning structures Pongchairerks et al. [15]. This version of PSO incorporates two additional terms to express social behavior. The PSO framework for solving the VRPTW is described in Figure 1.

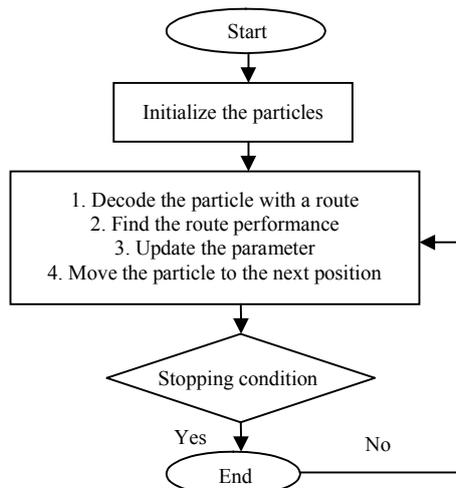


Figure 1: PSO flowchart for solving the VRPTW

3.1. Initialize the particles

In the initialize step, a swarm consisting of i particles is initialized by setting the position, velocity and personal best value of each particle. For each dimension of a particle, the position value (x_{id}) is set randomly in the range of the minimum and maximum position values, $x_{id} = U(x_{min}, x_{max})$, the velocity value is initially defined as zero ($v_{id} = 0$), and personal best value (p_d) is equal to the position value ($p_{id} = x_{id}$).

3.2. Main steps

Kennedy et al. [10] developed a non-iterative algorithm for the fitting of a sum of exponentials to first order data. The goal of the first order sum of exponentials fit is to fit a sum of exponentials to a vector r so that any element r_i ($i = 1, \dots, N$) in r follows the function. The main part of the PSO framework is the iteration step. As shown in Figure 1, each iteration step consists of four sub-steps, namely the decode particle to vehicle route, evaluate route performance, update cognitive and social term, and update particle velocity and position. In the first sub-step, each particle position is decoded to a set of vehicle routes by the decoding method that will be explained later. Then, the

performance of each constructed set of routes (i.e., the total travelled distance of the routes) is evaluated. This performance value is then kept as the fitness value of its corresponding particle. The mathematical notations of the PSO algorithm are defined as follows. Assume the searching space is D -dimensional and the total number of particles is n . The i -th particle location is denoted by vector $X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$; the past optimal location of the i -th particle in the "flight" history (i.e., the location corresponds optimal solution) is $P_i = (p_{i1}, p_{i2}, \dots, p_{iD})$. The past optimal location P_g of the g -th particle is optimal in all of $P_i (i=1, 2, \dots, n)$; the location changing rate (i.e., speed) of the i -th particle is denoted by vector $V_i = (v_{i1}, v_{i2}, \dots, v_{iD})$. $[x_{min}, x_{max}]$ is the changing range of the particle location. $[v_{min}, v_{max}]$ is the changing range of the speed. If the location and speed exceed boundary range in iteration, give the boundary value. Based on the analysis of the above parameters, Clerc [11] provided the parameter conditioning of the PSO convergence. In the last sub-step, the velocity of particle is updated based on the social and cognitive terms before the particle is moved to a new position based on its velocity according to Equations (1) to (3).

$$v_{id}(t+1) = w \times v_{id}(t) + c_p \times u \times [p_{id}(t) - x_{id}(t)] + c_g \times u \times [p_{gd}(t) - x_{id}(t)] + c_l \times u \times [p_{idL}(t) - x_{id}(t)] + c_n \times u \times [p_{idN}(t) - x_{id}(t)] \quad (1)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad (2)$$

$$w(t) = w_{end} + (I_{max} - I) \times (w_{start} - w_{end}) / I_{max} \quad (3)$$

where, u is a uniform distributed random number within interval $[0, 1]$; w is the inertia weight that linearly decreases and is within interval $[w_{start}, w_{end}]$; c_p , c_g , c_l and c_n are the acceleration constant of cognitive term, g_{best} social term, l_{best} social term and n_{best} social term, respectively. I_{max} is the maximum iteration and i is the current iteration. For controlling the search range of particles, if the position of a particle in a particular dimension exceeds the minimum or maximum position, then the position value is set on the bound and the corresponding velocity value is set to zero.

3.3. Stopping condition

The stopping condition controls the mechanism for the repetition of this algorithm. The algorithm is finished whenever the stopping condition is satisfied; otherwise, the iteration step is repeated. The stopping criterion used in this framework is the total number of iteration.

3.4. Solution representation and decoding method

The decoding method is the first part of the iteration step in the PSO framework for solving the VRPTW described in this section. The decoding method is responsible for transforming each specific particle representation into a set of vehicle routes, in which its performance is kept as the fitness value of the particle. An indirect representation of $(n + 2 \times m)$ -dimensional particle is proposed to represent the VRPTW solution with n customers and m vehicles (see Figure 2).

| | | | | | | | | | | |
|-------------------|---|-----|-----------|-------|-----------|-------|-------------|-----|------------------|----------------|
| 1 | 2 | ... | n | $n+1$ | $n+2$ | $n+3$ | $n+4$ | ... | $n+2 \times m-1$ | $n+2 \times m$ |
| Customer Priority | | | Vehicle 1 | | Vehicle 2 | | Vehicle m | | | |

Figure 2: Solution representation

The particle position in each dimension is represented by a real number, so that it can be applied in the framework based on a continuous version of PSO. The indirect representation is chosen since the vehicle routes cannot be directly represented by continuous numbers. The representation consists of two main parts. The first part of the representation is related to the customers, in which each customer is represented by a factor that can be used to set the priority of the customers. Hence, n dimensions of a particle are required to represent n customers. Since it is possible to get infinite array combination of n continuous numbers and one combination leads to one priority list, a diverse set of vehicle routes can be generated. The second part of the representation is related to the vehicles that incorporate the unique idea of vehicle reference point. Reference point of a vehicle is defined as a point in the service map that represents a certain area that is most likely be served by the vehicle.

The first step of decoding method is setting a priority list of customer based on the corresponding n dimensions of particle position. The priority list is constructed using rule of a small position value, in which the smaller the position value of a dimension, the higher is the priority given to the corresponding customer. In the second step, the corresponding $2 \times m$ dimensions are converted into the reference point of vehicles. The last decoding step is to construct routes based on the customer priority list and the vehicle priority matrix. Another effort to improve the solution quality of the route is to re-optimize the emerging route by using some improvement heuristic methods.

4. Computational Experiments

Computational experiments are conducted to evaluate the effectiveness of the proposed PSO using the benchmark data of Solomon [4] as the test cases. Our algorithm is implemented in MATLAB on a PC with Intel P4 2.2 GHz–4 GB RAM. The PSO parameters are listed in Table 1.

Table 1: Parameter setting of the proposed PSO.

| Parameter | Abr. | Value |
|--|-------|--|
| Number of particle | I | 100 |
| Number of iteration | T | 1000 |
| Number of neighbour | K | 5 |
| Inertia weight | w | linearly decreasing from 0.9 to 0.1 |
| Personal best position acceleration constant | c_p | 2 |
| Global best position acceleration constant | c_g | 2 |
| Constant of lbest social term | c_l | 1.5 |
| Constant of nbest social term | c_n | 1.5 |

4.1. Experiments data

In the experiment, the revised Solomon's benchmark [4] is used, in which there are three types of data sets. It is noted that the VRPTW benchmark data of Solomon [4] comprises of the high variety of problem situations, such as location of customers, vehicles characteristic, customers' demand and corresponding time windows, in which the variation is implied on the problem instance name. As it is shown in Table 2, the solution quality of PSO for 25-customer problems is good and consistent. Overall, the average results are very close to the optimal solution as indicated by the average deviation that is less than 1%. It is also shown from Table 2 that the proposed PSO yields good solutions with reasonable fast computational times. Moreover, consistency over replications is implied by the results. Almost all the results give the same results over five replications as shown by the same minimum, average and maximum values of the objective function. The results are also desirable, since the good solution quality is guaranteed to be reached in reasonable time.

Table 2: Computational results of the proposed PSO algorithm

| Problem | No. of Customers | CPU Time (Sec.) | PSO solution | Optimum solution | Error (%) |
|---------|------------------|-----------------|--------------|------------------|-----------|
| C101 | 3 | 35 | 191.8 | 191.3 | 0.26 |
| C102 | 3 | 33 | 190.6 | 190.3 | 0.16 |
| C103 | 3 | 34 | 190.7 | 190.3 | 0.21 |
| C104 | 3 | 38 | 192.1 | 186.9 | 2.78 |
| C105 | 3 | 39 | 191.8 | 191.3 | 0.26 |
| C201 | 2 | 57 | 215.5 | 214.7 | 0.37 |
| C202 | 2 | 53 | 215.4 | 214.7 | 0.33 |
| C203 | 2 | 54 | 215.5 | 214.7 | 0.37 |
| C204 | 2 | 89 | 213.9 | 213.1 | 0.38 |
| C205 | 2 | 59 | 215.5 | 214.7 | 0.37 |
| R101 | 8 | 32 | 618.3 | 617.1 | 0.19 |
| R102 | 7 | 31 | 548.1 | 547.1 | 0.18 |
| R103 | 5 | 28 | 454.7 | 454.6 | 0.02 |
| R104 | 4 | 29 | 418.1 | 416.9 | 0.29 |
| R105 | 6 | 30 | 531.5 | 530.5 | 0.19 |
| RC101 | 4 | 37 | 462.2 | 461.1 | 0.24 |
| RC102 | 3 | 36 | 352.7 | 351.8 | 0.26 |
| RC103 | 3 | 34 | 333.9 | 332.8 | 0.33 |
| RC104 | 3 | 32 | 307.1 | 306.6 | 0.16 |
| RC105 | 4 | 35 | 412.4 | 411.3 | 0.27 |

4.2. Real-case study

We study a problem that is encountered by Water and Waste Water Company in Tehran, which is responsible for providing enough safe potable water for about 14 million consumers. The system described in this paper is being used in a particular kind of the VRPTW for the Chlorine capsule distribution. The distribution areas are located in Tehran with about 730 square Km. It is critical that each 60 water reservoirs can get their Chlorine capsule in a proper time to disinfect the water. This case includes 60 customers that are the water reservoir and must be served with 15 available trucks. Every truck can deliver 10 Chlorine capsules. We attempt to find an optimal solution to satisfy the demand. We perform a survey on the actual information of the orders and demands. Our experiment

shows when a particular customer is served. We want to reduce the incurred costs. Table 3 shows the distance, hard time window and the Chlorine capsules demand of each reservoir.

Table 3: Distance and demand of a real-case study

| No. | X(m) | Y(m) | Earliest time | Latest time | Demand | No. | X(m) | Y(m) | Earliest time | Latest time | Demand |
|-----|---------|---------|---------------|-------------|--------|-----|---------|---------|---------------|-------------|--------|
| 1 | 1006.1 | 2187.3 | 2 | 6 | 2 | 31 | 2399.9 | 381.4 | 0 | 3 | 2 |
| 2 | 4461.8 | 3235.3 | 3 | 6 | 3 | 32 | -4636.2 | -3012.2 | 0 | 7 | 2 |
| 3 | -2327.6 | 1190.7 | 3 | 8 | 1 | 33 | 2003.8 | 576 | 0 | 6 | 1 |
| 4 | -3871.2 | 687.7 | 3 | 5 | 1 | 34 | 3493.2 | 180.8 | 3 | 4 | 1 |
| 5 | -1932.6 | -1867.7 | 3 | 7 | 2 | 35 | 2895.9 | 844.5 | 3 | 5 | 5 |
| 6 | -3089.9 | -139.6 | 0 | 6 | 1 | 36 | -3627.7 | -2082.4 | 0 | 5 | 1 |
| 7 | -2289 | -169 | 2 | 8 | 3 | 37 | 2707.3 | 237.8 | 4 | 8 | 2 |
| 8 | -3409.6 | -1831 | 3 | 5 | 3 | 38 | 5671.2 | -280.2 | 0 | 3 | 2 |
| 9 | -1223.4 | -2118.8 | 3 | 8 | 2 | 39 | -3126.2 | -1850.8 | 3 | 8 | 3 |
| 10 | 2487.8 | -1732.2 | 1 | 2 | 1 | 40 | -5529.7 | -3383.5 | 4 | 7 | 3 |
| 11 | -2538.7 | -1633.9 | 2 | 4 | 1 | 41 | -1656.8 | -1306.8 | 1 | 1 | 4 |
| 12 | -4216 | 3485.3 | 0 | 5 | 2 | 42 | 1850.6 | -1111.3 | 0 | 5 | 3 |
| 13 | 1346.8 | 1875.2 | 0 | 8 | 3 | 43 | 2947.1 | 1181.6 | 0 | 7 | 1 |
| 14 | 5489.3 | 1989.5 | 0 | 2 | 3 | 44 | -2849.7 | -1932 | 3 | 7 | 2 |
| 15 | 1668.7 | 710.3 | 2 | 8 | 2 | 45 | 2059.2 | -1091 | 2 | 5 | 2 |
| 16 | -2403.8 | 596.6 | 1 | 7 | 2 | 46 | -4838.7 | 1963.8 | 0 | 8 | 3 |
| 17 | -3727.1 | 1393.1 | 3 | 7 | 2 | 47 | -1293.9 | -1441.1 | 3 | 8 | 1 |
| 18 | -1140.4 | 3747.2 | 3 | 7 | 1 | 48 | -3895.3 | -3197.8 | 0 | 4 | 4 |
| 19 | 4856.3 | -1350.2 | 2 | 7 | 2 | 49 | -1568.3 | 1215.1 | 4 | 6 | 2 |
| 20 | 6694.6 | -705.1 | 4 | 7 | 1 | 50 | 713.6 | 3227.5 | 3 | 7 | 1 |
| 21 | 3290.8 | 509.1 | 0 | 7 | 2 | 51 | -2087.7 | 24.8 | 1 | 8 | 1 |
| 22 | 1921 | -2867 | 0 | 5 | 3 | 52 | -1259 | 1895.7 | 2 | 4 | 3 |
| 23 | 6451.2 | 1795.1 | 1 | 7 | 4 | 53 | 542.6 | 1604.8 | 0 | 8 | 2 |
| 24 | 5180.6 | 78.7 | 3 | 5 | 1 | 54 | -2026.4 | 1108 | 2 | 4 | 4 |
| 25 | -2727.1 | -1467 | 1 | 7 | 1 | 55 | -1917.8 | 1780.6 | 0 | 3 | 1 |
| 26 | -1227.1 | -4284.9 | 2 | 7 | 1 | 56 | -16.8 | 3064.5 | 1 | 4 | 3 |
| 27 | -2007.7 | -1349.1 | 1 | 3 | 3 | 57 | 2436.9 | 171.9 | 2 | 8 | 2 |
| 28 | -3742.7 | -2713 | 2 | 6 | 2 | 58 | -352.5 | 3171.6 | 0 | 7 | 2 |
| 29 | -627.5 | -1923.4 | 0 | 2 | 1 | 59 | -1175.4 | 3079.4 | 0 | 8 | 1 |
| 30 | 3021.6 | -1359.3 | 4 | 8 | 3 | 60 | 2433.1 | 3942.9 | 3 | 5 | 3 |

We solve this problem with the parameters given in the previous section and the optimum routes are as follows.

Vehicle 1: 0-59-58-18-12-27-0.

Vehicle 2: 0-52-55-50-48-0.

Vehicle 3: 0-46-0.

Vehicle 4: 0-3-49-17-54-0.

Vehicle 5: 0-7-51-16-6-4-32-0.

Vehicle 6: 0-8-36-28-44-11-25-0.

Vehicle 7: 0-60-13-1-53-0.

Vehicle 8: 0-41-5-56-0.

Vehicle 9: 0-47-39-40-0.

Vehicle 10: 0-9-29-26-0.

Vehicle 11: 0-15-30-19-0.

Vehicle 12: 0-14-2-23-0.

Vehicle 13: 0-33-35-21-43-24-0.

Vehicle 14: 0-31-34-37-57-20-38-0.

Vehicle 15: 0-45-42-22-10-0.

In overall, the results significantly make the changes on profitability and create a motivation for the company management to use this method. Note that the recent increase on energy prices can increase the motivation to use efficient transportation planning. By the use of such a system, a benefit of a 10% can be achieved in comparison with the system used before.

5. Conclusions

This paper has proposed a meta-heuristic method based on particle swarm optimization (PSO) for the vehicle routing problem with time windows (VRPTW) in a real-case study. The computational experiments have shown that the proposed PSO framework and solution representation were effective to solve the VRPTW. The route construction heuristics was capable of increasing the solution quality of the route. Also, the structure and mechanism of our proposed PSO could generate diverse solutions and consistently maintain or improve the best found solution. The experimental results have shown that the algorithm has greatly improved the solution quality in term of the number of customers served when compared with the previous methods. It was also very competitive in term of distance minimization. The real case application of the Chlorine capsule distribution for Water and Waste Water Company in Tehran has been satisfactory. Some aspects could further improve the performance of the proposed algorithm, such as parameter optimization and programming implementation. Although the PSO parameter set used in this paper came from some preliminary experiment, it might not be the best one. In the experiments, parameters have affected on the optimization result, such as the population size of the algorithm and number of iterations. The experiments have shown that with the increase of the population size, the optimization results became better and more stable. When the population size has reached to a certain level, the results of the optimization have stabilized and have not been changed longer significantly with the increase of the population size. By increasing the number of iterations, the result was getting better and searching the excellent solution to the increasingly high probability. In addition, the programming implementation of the proposed PSO may be further optimized. Since these efforts may yet contribute to the additional performance gains in both the solution quality and computational time, a further study on these aspects is still necessary. Some further research to apply the proposed PSO to other VRP variants should be carried out to show generality of the method. Since these variants differ from one another only on the specific problem constraints, the adjustment is only required in the constraint feasibility checking of the decoding method. However, the effectiveness of this idea needs further exploration.

References

1. Dantzig G, Ramser J., 1959, "The truck dispatching problem," *Management Science*, 6, 80-91.
2. Ellabib I., Otman A., Calamai P. 2002, "An experimental study of a simple ant colony system for the vehicle routing problem with time windows," *Lecture Notes in Computer Science* 2463/2002, 53-64.
3. Golden B.L., Assad A., 1986, "Perspectives on vehicle routing: Exciting new developments," *Operations Research* 34, 803-809.
4. Solomon M., 1987, "Algorithms for the vehicle routing and scheduling problems with time window constraints," *Operations Research* 35, 254-265.
5. Chiang W., Russell R., 1996, "Simulated annealing meta heuristics for the vehicle routing problem with time windows," *Annals of Operations Research* 63, 3-27.
6. Taillard E.D., Laporte G., Gendreau M. 1996, "Vehicle routing with multiple uses of vehicles," *Journal of the Operational Research Society* 47, 1065-1070.
7. Campbell A.M., Savelsbergh M. 2005, "Decision support for consumer direct grocery initiatives," *Transportation Science* 39, 313-327.
8. Kennedy, J. and Eberhart, R., 1995, "Particle swarm optimization. Proceedings of IEEE International Conference on Neural Networks, 4, 1942-1948.
9. Kennedy, J. and Eberhart, R., 2001, "Swarm intelligence," Morgan Kaufmann Publishers, USA.
10. Clerc, M., 2006, "Particle swarm optimization," ISTE, London, UK.
11. 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*, 7(4), 607-614.
12. Qing, Z., Limin, Q., Yingchun, L. and Shanjun, Z., 2006, "An improved PSO algorithm for VRPTW," *IEEE Congress on Evolutionary Computation*, Vancouver, Canada, 1386-1390.
13. Tasgetiren, M.F., Liang, Y.C., Sevkli, M. and Gencyilmaz, G., 2007, "A particle swarm optimization algorithm for makespan and total flowtime minimization in the permutation flowshop sequencing problem," *European Journal of Operational Research*, Vol. 177, No. 3, pp.1930-1947.
14. Alvarenga, G., Mateus, G. and Tomi, G., 2007, "A genetic and set partitioning two-phase approach for the vehicle routing problem with time windows," *Computers and Operations Research*, 34(6), 1561-1584.
15. Pongchairerks, P. and Kachitvichyanukul, V., 2009, "A two-level particle swarm optimization algorithm on job-shop scheduling problems," *International Journals of Operational Research*, 4(4), 390-411.