Available online at: http://e-jurnal.lppmunsera.org/index.php/JSMI

Jurnal Sistem dan Manajemen Industri

ISSN (Print) 2580-2887 ISSN (Online) 2580-2895

Implementation of discrete particle swarm optimization algorithm in the capacitated vehicle routing problem

Aisyahna Nurul Mauliddina[1*], Faris Ahmad Saifuddin[1], Adesatya Lentera Nagari[1], Anak Agung Ngurah Perwira Redi[2], Adji Candra Kurniawan[1], Nanda Ruswandi[1]

[1] Department of Logistics Engineering, Pertamina University, Jl. Teuku Nyak Arief, Keb. Lama, South Jakarta 12220, Indonesia
[2] Department of Industrial Engineering-Binus Graduate Program, Binus University, Kebon Jeruk, West Jakarta 11530, Indonesia

ARTICLE INFORMATION

Article history:

Received: September 30, 2020
Revised: December 8, 2020
Accepted: December 21, 2020

Keywords:

Discrete particle swarm optimization
Capacitated vehicle routing problem
Repeated measure ANOVA
Metaheuristic

*Corresponding Author

Aisyahna Nurul Mauliddina
E-mail: dinaaisyahna@gmail.com

ABSTRACT

Capacitated Vehicle Routing Problem (CVRP) is known as an NP-hard problem. It is because CVRP problems are very hard for finding optimal solutions, especially in large instances. In general, the NP-hard problem is difficult to solve in the exact method, so the metaheuristic approach is implemented in the CVRP problem to find a near-optimal solution in reasonable computational time. This research uses the DPSO algorithm for solving CVRP with ten instances of benchmark datasets. DPSO implementation uses tuning parameters with the One Factor at Time (OFAT) method to select the best DPSO parameters. The outcome objective function will be compared with several PSO models proposed in previous studies. Statistical test using One Way Reputed Measure ANOVA is needed to compare algorithm performance. First, ANOVA uses for comparing's results. Then, ANOVA is also used to test DPSO's performance compared with DPSO-SA, SR-1, and SR-2 algorithm. The computational result shows that the basic DPSO algorithm not competitive enough with other methods for solving CVRP.

1. INTRODUCTION

Vehicle Routing Problem (VRP) is a problem design closely related to logistics and distribution activities [1]. Capacitated Vehicle Routing Problem (CVRP) is one of the VRP problems first introduced by Dantzig and Ramser in 1959 [2]. CVRP is a design problem that has the objective function of minimizing the total cost of distribution from the depot to several locations to meet consumer demand. In the CVRP model, the distribution process uses several similar vehicles with a uniform capacity to deliver to consumers [3].

CVRP is a broad and general problem, so researchers develop a CVRP model to be more specific according to the problem characteristic. Some of the CVRP model developments include the aim of being able to handle several additional constraints such as the number of stochastic requests [4], time constraints or time windows [5], fuzzy demand constraints [6], etc. CVRP is considered the key to deal with operational problems related to determining vehicle routes. The difficulties in solving CVRP problems depend on the problem's complexities.

http://dx.doi.org/10.30656/jsmi.v4i2.2607

In terms of problem complexity, CVRP is an NP-hard problem. Solving the CVRP problem is often faced with the difficulty of obtaining optimal results and long computational times. Therefore, further research is needed on the right approach method to get optimal computation time results [7]. The number of points and constraints makes CVRP problems very difficult to solve using the exact method or conventional methods. Finding the optimal solution of NP-hard problems with large instances is usually using a metaheuristic algorithm approach. Several previous studies have developed a metaheuristic approach in solving CVRP problems, such as using genetic algorithms [8], particle swarm optimization (PSO) [9], tabu search [1], and several other algorithms. This study will further solve CVRP problems using the Particle Swarm Optimization (PSO) algorithm's basic model.

PSO is a population-based metaheuristic algorithm invented by Kennedy and Eberhart [10]. The PSO algorithm adapts the behavior of a group of birds or fish in nature by simulating the movement of individuals in the group to obtain optimal values. The PSO algorithm has several advantages, such as can be used for stochastic objective functions, can easily get out of the optimal locale, and simple in programming and implementation. We also do parameter tuning for finding optimal parameter values. Besides, the determination of initial solutions will not affect the optimal results obtained [4]. The PSO algorithm uses individual physical movements in the swarm. It has a flexible and balanced mechanism to be able to adapt to the best values obtained by individuals and swarms [9]. Initially, the PSO algorithm is intended for continuous problems, but several studies have developed the PSO algorithm applied to discrete problems. Research related to the application of Discrete Particle Swarm Optimization (DPSO) on CVRP problems has been carried out by Chen et al. [9], who combined DPSO with the Simulated Annealing (SA) algorithm and research by Ai and Kachitvichyanukul [7] which developed DPSO with two different representations of solutions.

The initial idea of the PSO algorithm is to solve the continuous optimization problem. Based on two main vectors' role, namely the position vector x_i and the acceleration vector v_i, whose values are always changing in continuous space. For discrete problems, the vector x_i is associate with a feasible route along with the total distance value. The biggest problem with the DPSO algorithm is how to formulate the acceleration vector v_i to fit the context of the routing problem.

Quite a lot of research has been developed in transforming PSO to solve problems in Discrete PSO problems. The transformation pattern has developed based on the basic idea of PSO, namely that the swarm members can move randomly, and tend to be in the best position in the past and consider probabilities. The advantage of this algorithm is its ease of use with fewer parameter tunning to adjust. DPSO has been successfully applied to a vast range's applications such as function optimization, neural network training, task assignment, and scheduling problem [11].

Several studies using the PSO algorithm to solve VRP problems include solving capacitated vehicle routing problems using PSO was investigated by Ai and Kachitvichyanukul [7], showing that the second solution representation (SR-2) is better than representation SR-1. Zou et al. [12] also researching solving multimodal-optimization problems using PSO. Their study results indicate that the close neighbor mobility optimization algorithm performs better than most single-objective multimodal algorithms. Singh and Singh [13] utilizes the PSO algorithm to solve transportation problems. The study results show that the performance of the proposed algorithm is statistically validated using the paired t-test.

This study will focus on solving CVRP problems with the basic DPSO algorithm using the data used by Chen et al. [9] in their research. This study will further examine the basic DPSO algorithm by setting its parameters in such a way that it can produce optimal values. This study's parameter tuning process is based on some literature and uses the One Factor at Time (OFAT) method to get the best parameters. Then, the best parameter uses for implementing the DPSO algorithm. The DPSO results in this problem will be compared to the two previous studies using statistical analysis. This study investigates whether the DPSO algorithm with parameter tuning is used in this problem competitive enough compared with Chen, SR-1, and SR-2 algorithms. So, the contribution of this research is not to propose a new algorithm, but this research used a basic DPSO algorithm without any hybridization or modifycation to show what is the DPSO standard with parameter setting can compete with Chen, SR-1, SR-2.

This study also provides statistical analysis for comparing the average of objective value between

http://dx.doi.org/10.30656/jsmi.v4i2.2607

basic DPSO with the other three algorithms. Although this research uses basic DPSO without modification or hybridization, this research is conducted with parameter tuning with several combinations. After that, the statistical analysis is carried out to prove the proposed algorithm's performance with the previous best-known solution.

2. RESEARCH METHODS

This section consists of the DPSO algorithm in the CVRP problem, parameter tuning methods, and statistical analysis used in this research.

2.1. CVRP Problem

There are several VRP problem variations, one of which is the Capacitated Vehicle Routing Problem (CVRP). In CVRP problems, there are additional constraints in the form of vehicle capacity. CVRP define as the problem of determining the optimal route in meeting consumer demand to obtain the optimal and minimum possible travel costs, travel times, and the number of vehicles [14]. The mathematical model of the CVRP problem is model as follows [15]:

The objective function of the CVRP mathematical model:

Minimize

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

Subject to:

$$X_{ij}^{k} = \begin{cases} 1, & \text{if vehicle k has a route from i to j} \\ 0, & \text{if nothing} \end{cases} \tag{2}$$

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

$$\sum_{k=1}^{K} \sum_{j=0}^{N} X_{ij}^{k} = 1, \qquad i = 1, 2, \dots, n \tag{4}$$

$$\sum_{i=0}^{N} X_{it}^{k} - \sum_{j=0}^{N} X_{tj}^{k} = 0, \qquad k = 1, 2, \dots, K; t = 1, 2, \dots, N \tag{5}$$

$$\sum_{j=0}^{N} q_{j} \left(\sum_{i=0}^{N} X_{ij}^{k} \right) \leq Q_{k}, \qquad k = 1, 2, \dots, K \tag{6}$$

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

$$\sum_{i=1}^{N} X_{i0}^{k} \leq 1, \qquad k = 1, 2, \dots, Ks \tag{8}$$

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

In the above mathematical model, the notation *N* represents the number of consumers, *K* is the number of vehicles used, is the distance traveled from consumer i to consumer j using vehicle *K*, and is the distance traveled from consumer i to consumer j using vehicle *K*. Additionally, the notation i, represents the capacity of the general. The detailed notation can be seen in Table 1.

The objective function of equation (1) is to minimize the total distance traveled by the vehicle. Constraints in equations (3) and (4) explain that each customer will only be visited by one vehicle. Equation (5) show that the number of vehicles leaving and returning to the depot is the same. Equation (6) explains that the cargo carried by the vehicle will not exceed the predetermined capacity. Equations (7) and (8) explain that the vehicle is only used once in a distribution cycle, and equation (9) is used to ensure that the variable taken is an integer number (0 or 1).

Table 1. Notation of mathematical model

Parameters	Description
C_{ij}^{k}	Distance traveled from depot i to agent j
X_{ij}^{k}	Vehicle k travels from depot i to agent j
q_j	Demand at agent j
Q_k	Carrying capacity of vehicle k
X_{ij}^{k}	Route of vehicle k start from depot 0

2.2. DPSO for Solving CVRP Problem

The PSO algorithm was first developed by Eberhart and Kennedy [10] imitated on swarm behavior in nature, such as a group of fish or birds [16]. Almost all cases of optimization, computational intelligence, and application design use the PSO algorithm. This algorithm is also usually combined with other algorithms according to the case's specifications to be solved. The PSO concept is usually implemented to non-linear optimization problems. Also, PSO can be used to problems that are very difficult to find the global optimum and get out of the optimal local trap [16].

The PSO algorithm has several parameters, including particles, particle velocity, personal best, and global best. The main components of the PSO algorithm are position and velocity [17]. The initial nodes of the particles in the population are determined randomly within the search area's boundary. Also, the objective value calculates from that point for each particle. The initial velocity is determined randomly within the specified velocity range. Based on each particle's objective value, each individual's best value can be determined, namely the Personal Best (P_{best}) and the best score in the group called Global Best (G_{best}). This research will use an optimizing model introduced by Eberhart and Shi [18] as follows:

$$V_{id} = w\ x\ V_{id} + c_1 \times rand \times (P_{best} - X_{id}) + c_2 \times rand \times (G_{best} - X_{id}) \quad (10)$$

$$X_{id} = X_{id} + V_{id} \quad (11)$$

In equation 10 and 11, V_{id} represents the velocity of a particle, X_{id} is the position of a particle, w is inertia weight, c_1 and c_2 represent the learning parameter. Meanwhile, $rand$ is a random number with a range 0-1.

The procedure for implementing the DPSO algorithm not different enough from the PSO algorithm. It begins by determining the initial location and velocity, determining other parameters, performing constraint handling, determining the Pbest and Gbest values, and updating the individual speed and position. This step is carried out until the termination criteria are reached [6]. The results obtained from the DPSO algorithm depends on the parameter values used [7]. These parameters include swarm size, learning of cognitive learning and social learning, inertia weight, etc. Therefore, DPSO implementation requires a parameter tuning process to determine the best parameter values to obtain optimal results. In order to get an optimal result, the fitness or objective value of the algorithm is evaluated by the following equation:

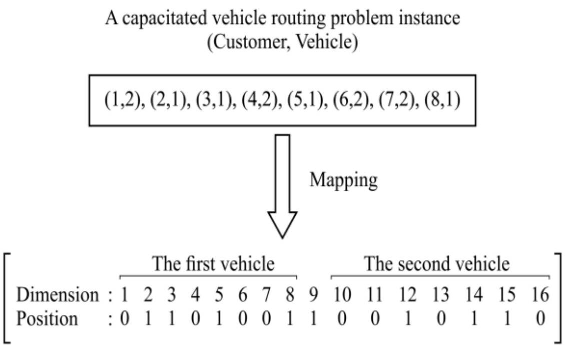
$$Objective\ Value = \sum_{k=1}^{K}\sum_{i=0}^{N}\sum_{j=0}^{N} c_{ij}x_{ij} \quad (12)$$

In general, the fitness function is to minimize the total cost. The DPSO algorithm's implementation begins with the initialization of particles and particle velocity in the swarm, setting parameters and constraint handling, calculating the fitness value, determining P_{best} and G_{best}, updating the velocity and location of particles. These steps repeat until they meet the termination criteria. The DPSO algorithm for CVRP problems in detail shown in Fig. 1. To implement the DPSO algorithm for the CVRP problem, the algorithm's solution follows previous research by Chen et al. [9].

For example, the CVRP problem has N number of customers that have to be served by K number of vehicles. So, there are $N \times K$ dimensions where each particle consists of K sections, and each section has N discrete points. As we know that the value of discrete points is 0 or 1, so value 1 represents that a customer is served by a vehicle. As shown in Fig. 2, there is a solution representation if we have eight customers and two vehicles in the CVRP problem.

```
Start
        Parameter Setup
        Initialize Swarm;
        Evaluate the objective function of each particle;
        Evaluate the objective function of each particle
        Repeat
        do
                for all particle i
                -     Calculate the new velocity of the particles
                -     Calculate the new position of the particle
                -     Constraint Handling
                -     Evaluate the objective function value of the particles
                -     Find new P Best and G Best particles
                -     If f(xi) < f(P Besti) Then P Besti = xi
                -     If f(xi) < f(G Besti) Then G Besti = xi
                -     Update the location and velocity of the particles
                -     Until Termination Criteria
                End for
                Calculate non-improvement
         Until Stopping Criteria
         Print Result
End
```

Fig 1. Basic DPSO's pseudocode



Dimension :	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Position :	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0

Fig 2. DPSO's solution representation

Because the problem is a CVRP problem, we have to ensure that the DPSO algorithm follows the VRP constraints. Every customer can only be served with one vehicle, and the length of each route is less than the constraint, and the total demand cannot be more than the capacity of vehicles.

2.3. Parameter Tuning

Several parameters are used in the DPSO algorithm, including swarm size, learning factors, maximum iterations, and inertia weight [19]. Determining the optimal parameter depends on the type of problem, and the standard rules are not determined. In this study, the parameter tuning

http://dx.doi.org/10.30656/jsmi.v4i2.2607

process used for choosing the following four parameters:

a. Inertia Weight (w)

The determination of the optimal inertia weight value to improve the DPSO algorithm's performance has been researched and developed. The inertia weight value affects the particles' velocity, determining the probability of moving and searching for particles. Shi and Eberhardt explained that the inertia weight value ranging from 0.9 to 1.2 could improve the DPSO algorithm's performance [18]. Other studies have shown an inertia weight value between 0.4-0.9 suitable for quite complicated problems [20].

b. Learning Factors

Learning factors in the DPSO algorithm include cognitive learning and social learning. Cognitive learning represents how much the P_{best} value contribution to calculating the velocity of particles. Meanwhile, social learning describes the influence of G_{Best} in calculating the velocity value. Particle velocity is important to pay attention to avoid premature convergence. The value of learning factors c1 = 2, c2 = 2 is recommended for obtaining optimal results [20]. For DPSO parameters, [18] used 0.5 for cognitive learning (c_1) and social learning (c_2), respectively. Another study by Wang et al. [21] used learning factor values ranging from 0.5 to 2.5.

c. Swarm Size

The determination of the number of particles directly affects computation speed and the speed of finding the optimal result. Several studies explain the technique of determining the number of particles. The optimal number of particles determine in the $5x_N - 10x_N$ range, where x_N is the number of variables [22]. Other studies have shown that the optimal number of particles is between 10-30 particles [23]. Meanwhile, Wang et al. [21] used the number of swarms varying from 25 to 125.

2.4. Statistical Analysis

This study uses statistical analysis to assess the selected DPSO parameters' performance and test the DPSO algorithm's performance compared to other algorithms. Statistical analysis of data includes descriptive statistics and hypothesis testing. Descriptive statistics are needed to see the details of the observed data. The hypothesis test results in this study prove the difference in the average value between the experimental data.

The hypothesis test used in this research is One Way Repeated Measure ANOVA. It aims to compare more than two average experiments with different treatments but use the same instance. Before doing the Repeated Measure One Way ANOVA test, it must do normality and the sphericity test. The normality test was carried out using the Kolmogorov-Smirnov test and the Saphiro-Wilkins test. While the sphericity test was carried out using Muchly's test.

3. RESULTS AND DISCUSSION

In this section, DPSO is used to solve basic CVRP problems by using benchmark data. The parameters used in the implementation of DPSO are the best parameters determined based on the tuning parameters using the OFAT method. The optimal results obtained from this algorithm will compare with the DPSO-SA, SR-1, and SR-2 algorithms.

3.1. Data

The dataset used in this research includes ten benchmark data used by Chen et al. [9] and Ai and Kachitvichyanukul [7], as shown in Table 2. The ten instances then solved using DPSO using Visual Studio applications and C# programming languages. In this research, simulations will carry out using a laptop with Intel Core i5 @ 1.8GHz quad-core with Turbo Boost-8GB RAM.

Table 2. Benchmark datasets

Instances	Customers	Vehicle
An33k5	32	5
An46k7	45	7
An60k9	59	9
Bn35k5	34	5
Bn45k5	44	5
Bn68k9	67	9
Bn78k10	77	10
En30k3	29	3
En51k5	50	5
En76k7	75	7

3.2. Parameter Tuning

The parameter setting in this study was carried out for four parameters, namely Inertia Weight (w), cognitive learning factor (c1), social learning factor (c2), and swarm size (N). The parameter tuning process begins with selecting the number of candidate values for each parameter.

http://dx.doi.org/10.30656/jsmi.v4i2.2607

After that, the determination of the optimal parameter value uses the OFAT method. The list of parameters in detail listed in Table 3. The selection of parameter value combinations is based on a literature study and refers to the parameters used in the study [21].

Table 3 listed the parameter combination for each experiment in every factor. The OFAT method changes the test variable's value and gives a fixed value for other variables. For example, when you want to determine the best value for c_1, we set a fixed value for inertia parameters weight, swarm size, and c_2. This study conducted four trials for inertia factors weight, four trials for factor c_1, three trials for factor c_2, and three experiments for the swarm size factor.

Each parameter's combination is implementted in DPSO the algorithm to determine the optimal value in all instances. Each experiment was run by replicating 30 times. The mean value for the objective function and the computation time is calculated for each experiment in each instance. After that, the experiment results were compared to see the effect of that parameter on the objective value and computation time for each parameter. The graph of the objective value and computation time for each factor can be seen in Fig. 3, Fig. 4, Fig. 5, and Fig. 6.

Table 3. Parameter value combinations

	Parameter				Trial
	w	c_1	c_2	N	
Inertia Weight (w)	0.9	2	1	25	Trial 1
	0.3	2	1	25	Trial 2
	0.4	2	1	25	Trial 3
	0.5	2	1	25	Trial 7*
Learning factor (c_1)	0.5	1	1	25	Trial 4
	0.5	1.5	1	25	Trial 5
	0.5	0.5	1	25	Trial 6
	0.5	2	1	25	Trial 7*
Learning factor (c_2)	0.5	2	1	25	Trial 7*
	0.5	2	1.5	25	Trial 8
	0.5	2	2	25	Trial 9
Swarm Size (N)	0.5	2	1	25	Trial 10
	0.5	2	1	50	Trial 11
	0.5	2	1	75	Trial 12

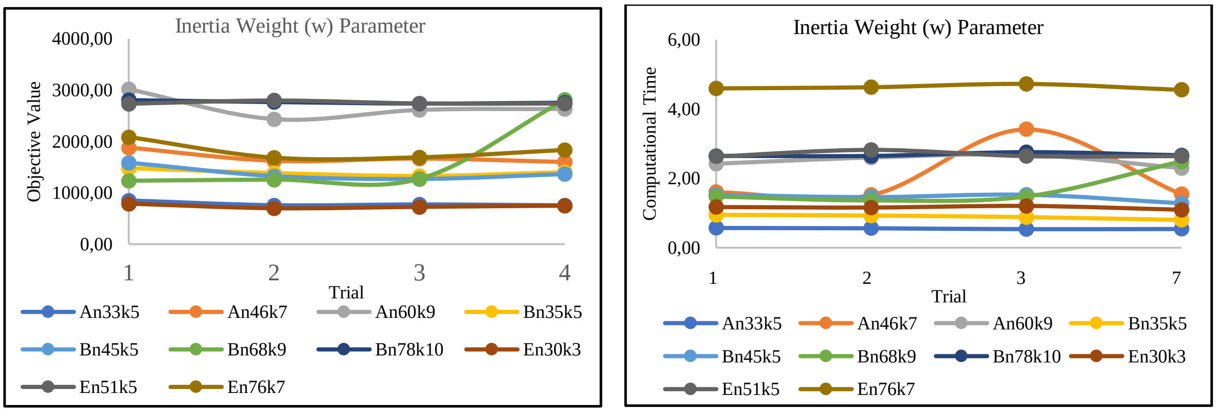


Fig 3. Comparison graph of objective values and computation time on inertia weight (w) factor

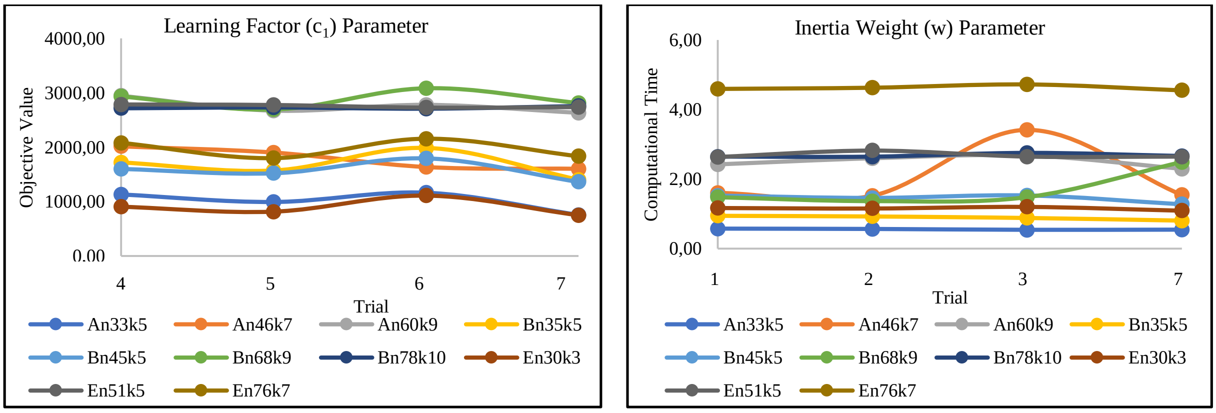


Fig 4. Comparison graph of objective values and computation time on learning factor (c_1)

http://dx.doi.org/10.30656/jsmi.v4i2.2607

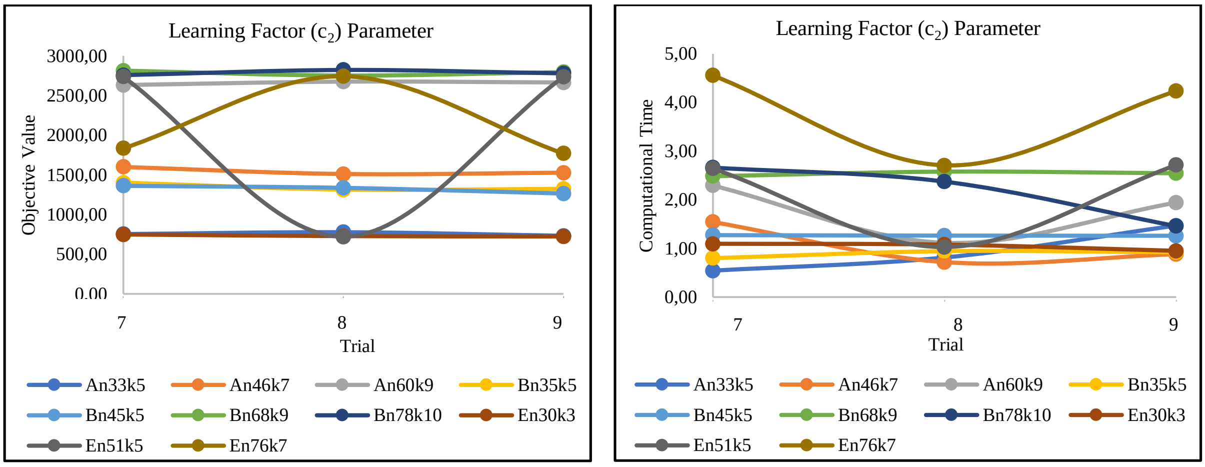


Fig 5. Comparison graph of objective values and computation time on learning factor (c_2)

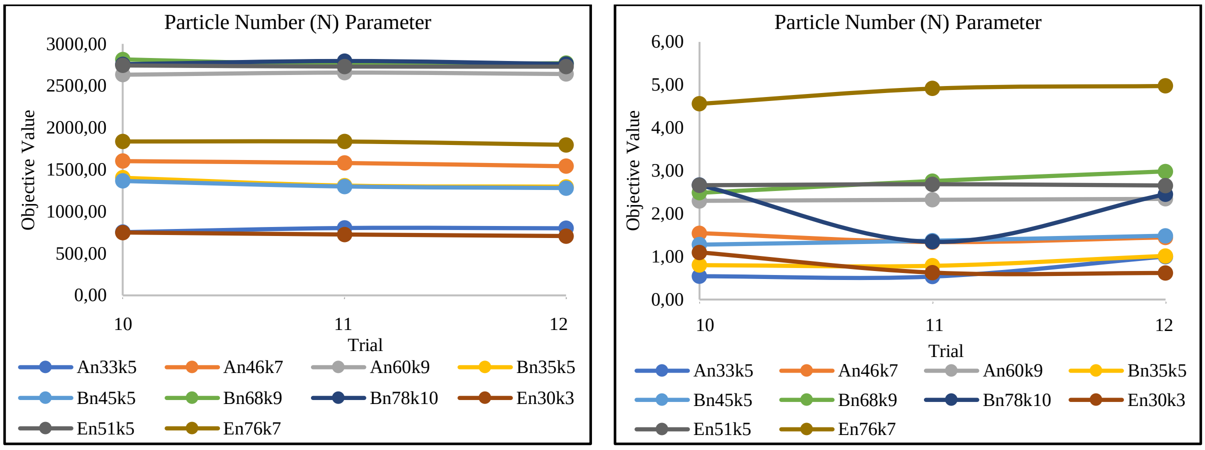


Fig 6. Comparison graph of objective values and computation time on particle number (N)

Fig. 3, Fig. 4, Fig. 5, and Fig. 6 shows that each parameter has a different objective value and computation time for each experiment. However, there is a similarity in the pattern for each experiment within a factor group for several instances. Besides, there are differences in the objective value and computation time between experiments that are not significantly different. So, it is difficult to determine whether there is a difference between the means. Therefore, a statistical analysis process using SPSS software is needed to see the differences in the respective experiments' value for each factor.

The next process is to perform statistical tests to determine the average value between the experiments in one factor. The statistical test used the One-way Repeated Measures ANOVA Test. This statistical test is needed because there are more than two experiments with the same instances that will be compared. Before conducting the Repeated Measures One Way ANOVA test, the first step is checking the assumption of the normality and homogeneity of variance. Furthermore, the Repeated Measures One Way ANOVA test was conducted with the following hypothesis:

H_0:$\mu_A = \mu_B = \cdots = \mu_n$; n = number of trials

H_1: at least one different experimental mean

The hypothesis above explains that when the significance value is less than the significance level = 0.05, the decision to be taken is to reject H_0, which means a difference in the average for one or more experiments. On the other hand, if the P-value significance is more than 0.05, then the decision taken fails to reject H_0, which means that there is no difference in the average between trials.

Repeated Measures One Way ANOVA was conducted for all four parameter factors. The results obtained are:

a. Inertia factor weight (w), there is no significant mean difference between trials 1, 2, 3, and 7.

http://dx.doi.org/10.30656/jsmi.v4i2.2607

b. In the cognitive learning factor (c_1), there is a significant difference in the average in several experiments. The mean difference was between experiments 4 and 5, experiments 4 and 7, experiments 6 and 7, and experiment 5 and 6.
c. Cognitive learning factor (c_2) shows no significant average difference between experiments 7, 8, and 9.
d. Swarm size factor, there is a difference in the average between the experiments conducted. The mean difference is found in experiments 10 and 12 and trials 11 and 12.

The results of the hypothesis test used for determining the best parameter value. The inertia weight factor and cognitive learning are not affected by changes in the value of the parameters. However, the best parameter is determined by considering the average optimal value and the average computation time required. The selected parameter is an experiment with parameters that can produce optimal results in an efficient time. On the inertia weight factor, the smallest objective value is reached by experiment 2 that equal to 1673.57, while the shortest computation time is obtained by experiment 1. However, because the difference in computation time between experiment 2 and experiment 1 is only 0.01, the selected parameter is experiment 2 with an inertia weight value of 0.3. However, for social learning factors (c_2) there is also no significant difference in mean. The lowest objective value was obtained in experiment 8 = 1738.43, while the fastest computation time is experiment 8 with an average time of 1.46. Therefore, the selected parameter was experiment 8 with the parameter value $c_2 = 1.5$.

Furthermore, the Repeated Measures One Way ANOVA test results show at least one different experimental mean value in cognitive learning parameters (c_1) and swarm size. However, hypothesis testing results can only conclude that there is at least one different average value. Therefore, the selected parameters' determination is based on the difference in the average experiment written in the Mean Difference (i-j) column in the Pairwise Comparison Table. Based on the table, the best cognitive learning parameters (c_2) is 1.5, while the swarm size is 50.

3.3. DPSO Implementation Result

The selected parameters that have been determined in the previous section are used in the implementation of the DPSO algorithm. Then, the DPSO uses to solve the ten instances written in Table 2. Each instance is replicated 10 times with parameter values w = 1.3, c1 = 1.5, c2 = 1.5, and N = 50. The computation is then done for the maximum computation time is the same as the SR-1 experiment by Ai and Kachitvichyanukul [7]. The DPSO experiment results will be compared with the DPSO-SA computation results by Chen et al. [9], SR-1, and SR-2 [7]. Among the three experiments, the average computation time generated by the SR-1 algorithm is the lowest compared to the other two algorithms. Therefore, the DPSO computation uses the SR-1 time to find out in the shortest time whether there is a significant difference in the mean of objective value from the DPSO algorithm and other algorithms. The computational results comparison between algorithms showed in Table 4.

Table 4. Computational result comparison

Instances	Objective Function				
	BKS*	**Chen**	**SR-1**	**SR-2**	**DPSO**
An33k5	661	661	661	661	661.354
An46k7	914	914	914	914	1554.885
An60k9	1354	1354	1366	1355	2246.501
Bn35k5	955	955	955	955	1271.618
Bn45k5	751	751	751	751	1319.581
Bn68k9	1272	1272	1278	1274	2409.634
Bn78k10	1223	1239	1239	1223	2487.445
En30k3	534	534	541	534	779.125
En51k5	521	528	521	521	950.238
En76k7	682	688	691	682	1481.925

*BKS: Best-known Solution

Table 4 shows that the Chen, SR-1, and SR-2 algorithms' objective values produce the same value for some instances and not significantly different. Meanwhile, the objective value generated by the DPSO algorithm with minimal computation time has not been able to achieve the best value from other algorithms. Furthermore, the metaheuristic performance test process is carried out using statistical tests to determine the difference in the mean value of the four algorithms and to find out which algorithm has a better resolution for the same data.

The statistical test used to analyze the algorithm's performance in this study was the Repeated Measure One Way ANOVA Test. This test was chosen because it aims to compare more than two algorithm models but use the same data instances. The variable that will be compared in statistical analysis is the total cost as the objective

http://dx.doi.org/10.30656/jsmi.v4i2.2607

value. Repeated Measure One Way ANOVA test can be done when the data meet the normally distributed assumptions and have equal variance. The result of the normality test showed in detail in Table 5.

Table 5. Normality test's result

		Kolmogorov-Smirnov[a]			Shapiro-Wilk		
	Alg.	**Stat.**	**df**	**Sig.**	**Stat.**	**df**	**Sig.**
Obj.	Chen	0.173	10	0.2*	0.902	10	0.231
	SR-1	0.174	10	0.2*	0.907	10	0.261
	SR-2	0.170	10	0.2*	0.908	10	0.270
	DPSO	0.177	10	0.2*	0.912	10	0.297

*. This is a lower bound of the true significance.
a. Lilliefors Significance Correction

The purpose of the normality test is to prove that the data have a normal distribution. The hypothesis in this normality test is as follows:

H_0: the data is normally distributed
H_A: the data is not normally distributed

Mauchly's Sphericity Test is a measurement used for validation on repeated measures ANOVA. Sphericity is an important assumption in the repeated measure ANOVA test, which measures whether there are differences in the variance of all possible pairs of independent variables. The hypothesis of Mauchly's Sphericity Test is as follows:

H_0: variance data is equivalent
H_A: variance data is not equivalent

Based on the data in Table 5, it can be seen that the significance value of each model for both the Kolmogorov-Smirnov Test and the Shapiro-Wilkin Test is more than 0.05. Therefore, the decision that can be taken is failing to reject H_0, which means proving the data for each model have a normal distribution. Furthermore, the ANOVA Repeated Measure Test using SPSS resulted in the following tables:

Table 6. Mauchly's sphericity test's result

Subject Effect	Mauchly	Approx. Chi-Square	df	Sig.	Epsilon[b]		
					Greenhouse-Geisser	Huynh-Feldt	Low-bound
Model	0	117.584	5	0	0.333	0.334	0.333

Test the null hypothesis that the orthonormalized transformed dependent variables' error covariance matrix is proportional to an identity matrix.
a. Design: Intercept
Within Subject Design: Model
b. May be used to adjust the degrees of freedom for the averaged test of significance.
Corrected test are displayed in the Test of Within-Subjects Effect table

Based on Table 6, it can be seen that the P-value is 0.000, which is less than the significance level of 0.05. Therefore, the decision that can be taken is to reject H_0, which means that the data variance is not equal. In the repeated measure ANOVA test, if the Mauchly's Sphericity Test results do not meet the equivalent variance assumption, then the determination of the repeated measure ANOVA test results are based on the Greenhouse-Geisser improvement value. Greenhouse-Geisser is a method used to improve the degree of freedom to reduce the Type I Error level. Decision making from a repeated-measure ANOVA test is based on the Tests of Within Subjects Effects that can be shown in Table 7**.** Test of within-subject effects is an important output in the Repeated Measures ANOVA Test. In this study, the Repeated Measures ANOVA test had the following hypothesis:

H_0: there is no difference average between the algorithm models
H_A: there is a difference in the average between the algorithm models

Table 7. ANOVA repeated measures test results

	Source	Type III sum of squares	df	Mean square	F	Sig.	Partial eta squared
Model	Sphericity assumed	2946685.131	3	982228.377	24.763	0	0.773
	Greenhouse-Geisser	2946685.131	1	2945515.768	24.763	0.001	0.773
	Hunh-feldt	2946685.131	1.001	2945118.659	24.763	0.001	0.773
	Lower-bound	2946685.131	1	29446685.130	24.763	0.001	0.773
Error (model)	Sphericity assumed	1070954.622	27	39664.986			
	Greenhouse-Geisser	1070954.622	9.003	118948.948			
	Hunh-feldt	1070954.622	9.005	118931.700			
	Lower-bound	1070954.622	9	118994.958			

http://dx.doi.org/10.30656/jsmi.v4i2.2607

As we know, the basis for decision making repeated measures ANOVA in this study refers to the Greenhouse-Geisser value. Table 7 shows that the significance value is 0.001, which means that the decision that can be taken is to reject H_0. Therefore, it can be concluded that based on the repeated measures ANOVA test, there are mean differences between the algorithm models tested. Furthermore, to determine the average of different models and determine which model with the better average value can finish by looking at the Pairwise Comparison, as shown in Table 8.

Table 8. Post-hoc analysis

(I) model	(J) model	Mean Difference (I-J)	Std. Error	Sig.[b]	95% Confidence Interval for Difference.[b]	
					Lower Bound	Upper Bound
1	2	-2.1	1.643	1	-7.627	3.427
	3	2.6	1.759	1	-3.317	8.517
	4	-626.632*	125.861	0.005	-1050.05	-203.211
2	1	2.1	1.643	1	-3.427	7.627
	3	4.7	1.832	0.183	-1.464	10.864
	4	-624.532*	125.383	0.005	-1046.35	-202.717
3	1	-2.6	1.759	1	-8.517	3.317
	2	-4.7	1.832	0.183	-10.864	1.464
	4	-629.232*	126.589	0.005	-1055.13	-203.332
4	1	626.632*	125.861	0.005	203.211	1050.053
	2	624.532*	125.383	0.005	202.717	1046.347
	3	629.232*	126.598	0.005	203,332	1055.132

Based on estimated marginal means

*. The mean difference is significant at the 0.05 level.

a. Adjustment for multiple comparisons: Bonderroni.

Based on Table 8, it can be seen in the significance column, which proves that there is an average difference in the Chen-DPSO model pair, the SR1-DPSO model, the SR2-DPSO model. Then to find out which average is different and which model has a better average value than another model that can see in the Mean Difference (I-J) column. This pairwise comparison was obtained from a Post-hoc analysis using the Bonferroni Procedure method. Table 8 shows that the Chen, SR1, and SR2 models are better than the DPSO models. Besides, the SR2 algorithm has better performance than the Chen and SR1 algorithms.

Based on Post-hoc results, further research may consider modifying the representation solution of the basic DPSO algorithm to improve the quality of the resulting solution. Besides, further research can use the hybridization of metaheuristic algorithms to obtain a better solution. The computation results using DPSO show that the average obtained with the minimum time has not been able to match the other three algorithms. The optimal objective value using DPSO only shows similarity to the best objective value ever reported in one instance, namely A-n33-k5. It means the basic DPSO algorithm needs a long computational time to find the best optimal solution compared with other algorithms. The statistical test results show that there are differences in the average objective value of each algorithm. So, based on the Post-hoc analysis, it can be concluded that the basic DPSO algorithm has not performed well enough when compared to the Chen, SR-1, and SR-2 algorithms. Besides, the SR2 algorithm has better performance than the Chen and SR1 algorithms.

4. CONCLUSION

This study implements the DPSO algorithm to solve CVRP problems. This research uses the OFAT method for selecting optimal parameter values. Based on OFAT's results, the best combination PSO parameters are w = 0.3, $c_1 = 1.5$, $c_2 = 1.5$, and N = 50. After that, the combination parameter is implemented in the DPSO algorithm to get each instance's best objective. After that, the DPSO algorithm's performance was compared with the Chen, SR-1, and SR-2 models from previous research. The computation results using DPSO show that the average obtained with the minimum time has not competed with the other three algorithms. The optimal objective value using DPSO only shows similarity to the best objective value ever reported in one instance, namely A-n33-k5. It means the basic DPSO with parameter setting the only algorithm that needs more computational time to find the best optimal solution than other algorithms. The statistical test results show that there are differences in the average objective value of each algorithm. Based on the Post-hoc analysis, it can be concluded that the basic DPSO algorithm has not competitive enough when compared to the Chen, SR-1, and SR-2 algorithms.

Therefore, it proved that basic DPSO needs more computational time to have competitive results than Chen, SR-1, and SR-2. For further research, or if we want to use DPSO for real problems, it needs more computational time to solve a problem using basic DPSO with parameter setting only. But, to solve a problem with efficient computational time, using DPSO with any hybridization or modification that matches the problem is better enough than using basic DPSO. Besides, depends on the post-hoc analysis, the SR2 algorithm has better performance than the Chen and SR1 algorithms.

http://dx.doi.org/10.30656/jsmi.v4i2.2607

REFERENCES

[1] P. Augerat, J. M. Belenguer, E. Benavent, A. Corberán, and D. Naddef, "Separating capacity constraints in the CVRP using tabu search," *Eur. J. Oper. Res.*, vol. 106, no. 2, pp. 546–557, 1998, doi: 10.1016/S0377-2217(97)00290-7.

[2] G. B. Dantzig and J. H. Ramser, "The Truck Dispatching Problem," *Manage. Sci.*, vol. 6, no. 1, pp. 80–91, Oct. 1959, doi: 10.1287/mnsc.6.1.80.

[3] Y. Marinakis, M. Marinaki, and G. Dounias, "A hybrid particle swarm optimization algorithm for the vehicle routing problem," *Eng. Appl. Artif. Intell.*, vol. 23, no. 4, pp. 463–472, 2010, doi: 10.1016/j.engappai.2010.02.002.

[4] Y. Marinakis, G.-R. Iordanidou, and M. Marinaki, "Particle Swarm Optimization for the Vehicle Routing Problem with Stochastic Demands," *Appl. Soft Comput.*, vol. 13, no. 4, pp. 1693–1704, 2013, doi: 10.1016/j.asoc.2013.01.007.

[5] M. A. Hannan, M. Akhtar, R. A. Begum, H. Basri, A. Hussain, and E. Scavino, "Capacitated vehicle-routing problem model for scheduled solid waste collection and route optimization using PSO algorithm," *Waste Manag.*, vol. 71, pp. 31–41, 2018, doi: 10.1016/j.wasman.2017.10.019.

[6] R. J. Kuo, F. E. Zulvia, and K. Suryadi, "Hybrid particle swarm optimization with genetic algorithm for solving capacitated vehicle routing problem with fuzzy demand – A case study on garbage collection system," *Appl. Math. Comput.*, vol. 219, no. 5, pp. 2574–2588, 2012, doi: 10.1016/j.amc.2012.08.092.

[7] T. J. Ai and V. Kachitvichyanukul, "Particle swarm optimization and two solution representations for solving the capacitated vehicle routing problem," *Comput. Ind. Eng.*, vol. 56, no. 1, pp. 380–387, 2009, doi: 10.1016/j.cie.2008.06.012.

[8] T. Xiao and Z. Fu, "A genetic algorithm for the open vehicle routing problem with soft time windows," *J. Railw. Sci. Eng.*, vol. 5, no. 2, pp. 79–83, 2008. Available: http://en.cnki.com.cn/Article_en/CJFDTotal-CSTD200802017.htm.

[9] A. L. Chen, G. K. Yang, and Z. M. Wu, "Hybrid discrete particle swarm optimization algorithm for capacitated vehicle routing problem," *J. Zhejiang Univ. Sci.*, vol. 7, no. 4, pp. 607–614, 2006, doi: 10.1631/jzus.2006.A0607.

[10] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, 1995, pp. 39–43, doi: 10.1109/MHS.1995.494215.

[11] A. R. Guner and M. Sevkli, "A Discrete Particle Swarm Optimization Algorithm for Uncapacitated Facility Location Problem," *J. Artif. Evol. Appl.*, vol. 2008, pp. 1–9, Apr. 2008, doi: 10.1155/2008/861512.

[12] J. Zou, Q. Deng, J. Zheng, and S. Yang, "A close neighbor mobility method using particle swarm optimizer for solving multimodal optimization problems," *Inf. Sci. (Ny).*, vol. 519, pp. 332–347, 2020, doi: 10.1016/j.ins.2020.01.049.

[13] G. Singh and A. Singh, "A hybrid algorithm using particle swarm optimization for solving transportation problem," *Neural Comput. Appl.*, vol. 32, no. 15, pp. 11699–11716, 2020, doi: 10.1007/s00521-019-04656-1.

[14] S. Irnich, P. Toth, and D. Vigo, "Chapter 1: The Family of Vehicle Routing Problems," in *Vehicle Routing*, Philadelphia, PA: Society for Industrial and Applied Mathematics, pp. 1–33, 2014, doi: 10.1137/1.9781611973594.ch1.

[15] S. Raff, "Routing and scheduling of vehicles and crews: The state of the art," *Comput. Oper. Res.*, vol. 10, no. 2, pp. 63–211, 1983, doi: 10.1016/0305-0548(83)90030-8.

[16] X.-S. Yang, "7 Particle Swarm Optimization," 2014, doi: 10.1016/B978-0-12-416743-8.00007-5.

[17] G. Lindfield and J. Penny, "Particle Swarm Optimization Algorithms," *Introd. to Nature-Inspired Optim.*, pp. 49–68, 2017, doi: 10.1016/b978-0-12-803636-5.00003-7.

[18] R. C. Eberhart and Y. Shi, "Particle swarm optimization: Developments, applications and resources," *Proc. IEEE Conf. Evol. Comput. ICEC*, vol. 1, no. February, pp. 81–86, 2001, doi: 10.1109/cec.2001.934374.

[19] M. Isiet and M. Gadala, "Sensitivity analysis of control parameters in particle swarm optimization," *J. Comput. Sci.*, vol. 41, p. 101086, 2020, doi: 10.1016/j.jocs.2020.101086.

[20] M. Pant, R. Thangaraj, and A. Abraham, "Particle swarm optimization: Performance tuning and empirical analysis," *Stud. Comput. Intell.*, vol. 203, pp. 101–128, 2009, doi: 10.1007/978-3-642-01085-9_5.

[21] H. Wang, Q. Geng, and Z. Qiao, "Parameter tuning of particle swarm optimization by using Taguchi method and its application to motor design," *ICIST 2014 - Proc. 2014 4th IEEE Int. Conf. Inf. Sci. Technol.*, no. 20130415, pp. 722–726, 2014, doi: 10.1109/ICIST.2014.6920579.

[22] J. S. Arora, *Introduction to Optimum Design*. Elsevier Science, 2004. Available: https://books.google.co.id/books?id=9FbwVe577xwC.

[23] I. C. Trelea, "The particle swarm optimization algorithm: Convergence analysis and parameter selection," *Inf. Process. Lett.*, vol. 85, no. 6, pp. 317–325, 2003, doi: 10.1016/S0020-0190(02)00447-7.