IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 12, No. 2, June 2023, pp. 656~666
ISSN: 2252-8938, DOI: 10.11591/ijai.v12.i2.pp656-666

An improved artificial bee colony with perturbation operators in scout bees' phase for solving vehicle routing problem with time windows

Salah Mortada[1], Yuhanis Yusof[2]
[1]Department of Computer Technology Engineering, Shatt Al-Arab University College, Basra, Iraq
[2]School of Computing, College of Arts and Sciences, Universiti Utara Malaysia, Kedah, Malaysia

Article Info

Article history:

Received Apr 22, 2022
Revised Sep 15, 2022
Accepted Oct 15, 2022

Keywords:

Artificial bee colony
Metaheuristic
Optimization
Swarm intelligence

ABSTRACT

An example of a combinatorial problem is the vehicle routing problem with time windows (VRPTW), which focuses on choosing routes for a limited number of vehicles to serve a group of customers in a restricted period. Meta-heuristics algorithms are successful techniques for VRPTW, and in this study, existing modified artificial bee colony (MABC) algorithm is revised to provide an improved solution. One of the drawbacks of the MABC algorithm is its inability to execute wide exploration. A new solution that is produced randomly and being swapped with best solution when the previous solution can no longer be improved is prone to be trapped in local optima. Hence, this study proposes a perturbed MABC known as pertubated (P-MABC) that addresses the problem of local optima. P-MABC deploys five types of perturbation operators where it improvises abandoned solutions by changing customers in the solution. Experimental results show that the proposed P-MABC algorithm requires fewer number of vehicles and least amount of travelled distance compared with MABC. The P-MABC algorithm can be used to improve the search process of other population algorithms and can be applied in solving VRPTW in domain applications such as food distribution.

Corresponding Author:

Salah Mortada
Department of Computer Technology Engineering
Shatt Al-Arab University College, Basra, Iraq
Email: salah.m.shaheen@sa-uc.edu.iq

1. INTRODUCTION

Logistics management is critical for lowering costs and increasing a company's competitiveness. Logistics firms must make strategic and operational choices to plan business operations effectively. Numerous businesses optimize truck routes to decrease costs and enhance the quality of logistic services. The vehicle routing problem (VRP) has shown promising results in chain management and other domains. The VRP may be described as the process of finding the most efficient set of vehicle routes to improve logistic businesses' competitiveness by saving time and cost [1], [2]. Thus, research on a variety of VRP variants were performed (e.g., VRP with pick-up and delivery (VRPPD), multi-depot vehicle routing problem (MDVRP) with time windows (MD-VRPTW), multiple depot VRP (MDVRP), and capacitated VRP (CVRP), and vehicle routing problem with time windows (VRPTW) [3], [4]. The VRPTW is an NP-hard issue that involves finding the optimal route for a group of limited-capacity vehicles between a central warehouse and a number of scattered customers, all of whom must be reached within a certain timeframe the time window as shown in Figure 1. VRPTW has been implemented in various real-world applications, including food distribution [5], container transport [6], perishable delivery [7], newspaper delivery [8], and petrol station [9]. In 1987, Solomon [10] first introduced heuristics to solve the VRPTW, and metaheuristics has become popular in recent years [11], [12].

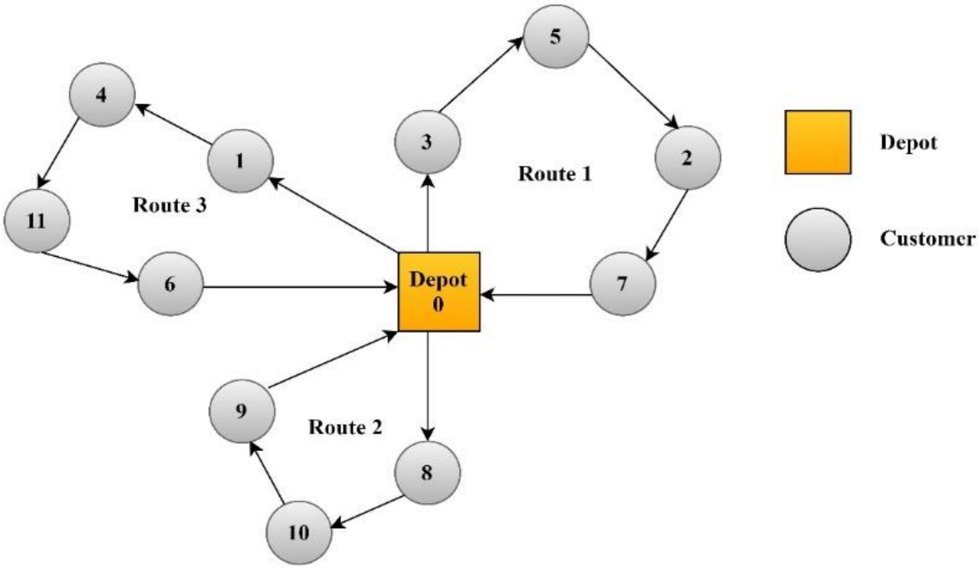

Figure 1. An example of VRPTW

Metaheuristics algorithms can be divided into local search and population-based techniques [13], [14]. The local search technique manipulates a single solution by exchanging segments of its components to produce better solutions, whereas the population-based technique uses more than one solution. Iterated local search and guided local search algorithms belong to the former class, whereas evolutionary computation, particle swarm optimization, grey wolf optimization and ant colony optimization belong to the later one [15], [16]. Population-based techniques are divided into two types, i.e., swarm intelligence algorithms and evolutionary algorithms [17], [18], on the basis of natural events that the algorithms represent. The theory of evolution is used by evolutionary algorithms to generate new species [19], [20]. Swarm intelligence algorithms rely on metaheuristics that mimic the collective behavior of problem-solving processes in self-organized systems [21], [22]. The collective intelligence emerges from the interactions of agents in social colonies with their surroundings [23], [24]. Metaheuristics algorithms have inspired many researchers to develop algorithms for VRPTW, including the genetic algorithm [25], particle swarm optimization [26], memetic algorithm [27], and the artificial bee colony (ABC) [28]. This research focuses on the ABC algorithm.

In 2005, Karaboga [29] developed the ABC algorithm to address numerical optimisation problems. Alzaqebah *et al.* [30] presented an updated version of the ABC algorithm for solving the VRPTW, and demonstrate that the modified ABC method beats the original. The bees in ABC algorithm are divided into three groups, namely, employed (EBs), onlooker (OBs), and scout (SBs) bees, adding to the bee colony's collective intelligence. The ABC algorithm is capable of exploration and exploitation. SB explores the search space globally and EB and OB explore the search space locally. In this research, the existing modified ABC (MABC) algorithm [30] is revised to solve the VRPTW. Although MABC is reported to be successful, its exploration process cannot execute wide exploration because a new solution is produced randomly and being swapped with best solution when the previous solution can no longer be improved. Randomly generated solutions, such as making it difficult to discover promising regions by exploring the area blindly, may affect the search process. SBs only focus on searching in relative proximity to the best solution, that is, exploiting around the same regions.

In this research, a new approach, perturbation of MABC (P-MABC) algorithm, is proposed to improve the exploration mechanism of the MABC. This approach uses five types of perturbation operators. These operators are adapted to perturb abandoned solutions by changing customers in the solution. This approach is performed to improve the exploration task and find many sub-ideal solutions or global ideal solution. Experimental results show that the proposed P-MABC approach improves the results. Overall comparison indicates that the P-MABC approach can obtain the best results in comparison with other optimization approaches, as presented by reducing the distance travelled and the number of vehicles, which are the main objectives of the VRPTW.

The remainder of this paper is organized as follows. Section 2 describes the mathematics of VRPTW. Section 3 offers the methods. Section 4 presents the proposed P-MABC approach. Section 5 explores the experimental results, performance evaluation, and benchmark datasets used in this research. Finally, section 6 provides the summary and the outlook for future research.

2. VEHICLE ROUTING PROPLEM WITH TIME WINDOWS

The VRPTW formulation, consisting of a single warehouse, a selected number of distributed customers, a number of homogeneous vehicles, and a network linking all customers to the warehouse, is presented in this section. The VRPTW benchmark is suggested by Solomon [10] and has 56 datasets, in which a predetermined set of vehicles must serve 100 customers. A connection between two nodes is seen by each arc in the undirected graph, and the direction of travel is defined. Any vehicle must begin from the warehouse, travel to the identified locations, and then return to the warehouse. Hence, each vehicle displays one route in the undirected graph. Based on the Solomon dataset, costs (c_{ij}) and time (t_{ij}) of travel are related to each arc in the undirected graph. The vehicle spends one unit of time for one unit of travelled distance.

Every vehicle has the same capacity, and only one of the vehicles must meet each customer once and have a defined deman. The overall capacity of all demands must be equivalent to or less than the full capacity of the vehicle operating on the route, and overloaded vehicles should not be present. A predetermined time interval, i.e., the earliest and latest times of access, shows the time window restrictions. Thus, the vehicle must reach the customer before the latest deadline for access and the vehicle must wait before the earliest access deadline. The service time needed for each customer to load/unload is often calculated and is specific irrespective of the demands' size. The goal is to establish a practicable route schedule, which reduces the total travel distance and, the number of vehicles. The VRPTW formulation is represented,

$$min \sum_{i=0}^{n} \sum_{j=0}^{n} \sum_{k=1}^{v} t_{ij} \times X_{ij}^{k} \quad (1)$$

$$min \sum_{i=0}^{n} y_{0k}$$

were,

$$X_{ij}^{k} = \begin{cases} 1 & \text{if vehicle } k \text{ travels from } c_i \text{ to } c_j \\ 0 & \text{otherwise} \end{cases}$$

$$y_{i}^{k} = \begin{cases} v & \text{if customer } c_i \text{ is served by vehicle } k \\ n & \text{otherwise} \end{cases}$$

$$\sum_{i=0}^{n} X_{ij}^{k} = y_{j}^{k}, \forall k = 1, \dots, v, \forall = 1, \dots., n \quad (2)$$

$$\sum_{j=0}^{n} X_{ij}^{k} = y_{i}^{k}, \forall k = 1, \dots, v, \forall = 1, \dots., n \quad (3)$$

$$\sum_{i=0}^{n} y_{i}^{k} \times q_i \le Q_k, \forall k = 1, \dots., v \quad (4)$$

$$\sum_{k=1}^{v} y_{i}^{k} = 1, \forall i = 1, \dots., n \quad (5)$$

$$\sum_{k=1}^{v} y_{0}^{k} = v \quad (6)$$

$$t_i + W_i + s_i + t_{ij} = t_j, \forall_i, = 0, 1, 2 \dots . n \neq j \quad (7)$$

$$e_i \le t_i \le l_i, \forall_i = 0, 1, 2 \dots . n \quad (8)$$

$$W_i = max\, \{e_i - t_i, 0\}, \forall_i = 0, 1, 2 \dots . n \quad (9)$$

Constraints (2) and (3) validates that each vehicle start and end from any customer until it finishes servicing the customer. Constraint (4) validates that the vehicle's capability is unviolated. Constraint (5) confirms the service is fulfilled only once for each customer. Constraint (6) ensures that the starting point of each vehicle is from the depot. Constraints (7)-(9) reflect time limitations to confirm that no time window is surpassed.

3. METHOD

The MABC algorithm [30]via SB lacks the ability to execute a wide exploration when solutions can no longer be improved, thus decreasing swarm diversity. This research improves the diversification in a promising region by exploring the search space to find the optimal solutions by using five perturbation operators and overcome this drawback. These operators are adapted to perturb abandoned solutions by changing customers in the solution. Such an approach is introduced as abandoned solutions and may contain beneficial information.

The five perturbation operators are explained as follow: (i) Forward insertion: Randomly select two customers (c_i and c_j) from the route and insert c_i after the position of c_j. (ii) Backward insertion: Randomly select two customers (c_i and c_j) from the route and insert c_j before the position of c_i. (iii) Two interchanges: The first step is to swap two customers randomly from separate routes. Next, reverse location, and then swap one customer from one route with one from the other. The purpose of this procedure is to minimize the distance by moving customers that may not be in the ideal route. (iv) Cross-exchange: This procedure cuts two separate routes into two sections and regroups them with crossing arcs. It swaps the segments from route r_1 and r_2 beginning at customer c_i and c_j sequentially. (v) 2 opt: This is an intra-route procedure that overturns a part of a route by removing two arcs randomly and switching them with the other two arcs to reform the route to minimize the operational expense.

4. PROPOSED P-MABC FOR VRPTW

The P-MABC algorithm starts with population initialization followed by the use of neighbourhood operations in EB and OB, and the use of perturbation operations by SB. This procedure is repeated until the termination condition is met. The main phases of the proposed P-MABC algorithm are as follows:

Initialization phase: By validating the constraints of VRPTW. The algorithm's initial population solutions are generated at random. Afterwards, each solution's fitness value is calculated, and the best solution is determined.

Exploitation process of employed bees (EB) phase: Solutions from the population are randomly allocated to each EB. Then, given the chosen solution in its memory, each EB tries to enhance the solultion by using a random neighborhood operation same as described previously. The existing solution is adjusted on the basis of the amount of nectar (i.e., fitness value distance) of the new solution. Each solution undergoes the neighborhood operation to investigate the surrounding area and improve the existing solution.

Exploitation process of onlooker bees (OB) phase: Based on the information given by the EB. Each OB is responsible for selecting potential food sources. Food sources are chosen on the basis of probability, which is calculated using (10),

$$p = \frac{fi}{\sum_{i=1}^{N} fi} \tag{10}$$

where fi is the fitness value of the i solution, and N is the number of food sources in the colony. A good solution is that with high probability (p) of the i food source. Each OB modifies the food source on the basis of the amount of nectar by performing the random neighborhood operation on the current solution in its memory (i.e., fitness value distance). Each solution undergoes the neighborhood operation to investigate the surrounding area and improve the existing solution. If the existing solution remains unchanged after a certain number of iterations, called the limit, the solution is intended to be abandoned, and OB is transformed into SB.

Exploitation process of scout bees (SB) phase: The SB phase is implemented to replace an unchanged solution called the abandoned solution after a certain number of iterations. If the number of trials for a food source is higher than a predefined value limit, SB generates a new solution on the basis of five moves i.e., forward insertion, backward insertion, cross-exchange, 2 opt, and two interchange, to perturb abandoned solutions instead of starting the search from a random solution or swap the generated one with best solution, P-MABC perturbs abandoned solutions for swarm diversity to discover the less-crowded region in the existing archive and possibly achieve non-dominated solutions.

Termination process: termination criterion of P-MABC. If the termination criteria are reached, P-MABC comes to a halt and uses the best solution identified so far. Moreover, EB, OB, and SB stages are repeated. The flowchart of the P-MABC algorithm is Figure 2 as shown in Appendix.

5. EXPERIMENTAL RESULTS AND COMPARISONS

In this section, an extensive set of experiments have been performed to assess the performance of P-MABC for VRPTW. Section 5.1 presents the benchmark dataset and the deployed parameter setting, while

Section 5.2 provides the analysis of experimental results. Focus of the comparison is to determine if the proposed P-MABC outperform the MABC.

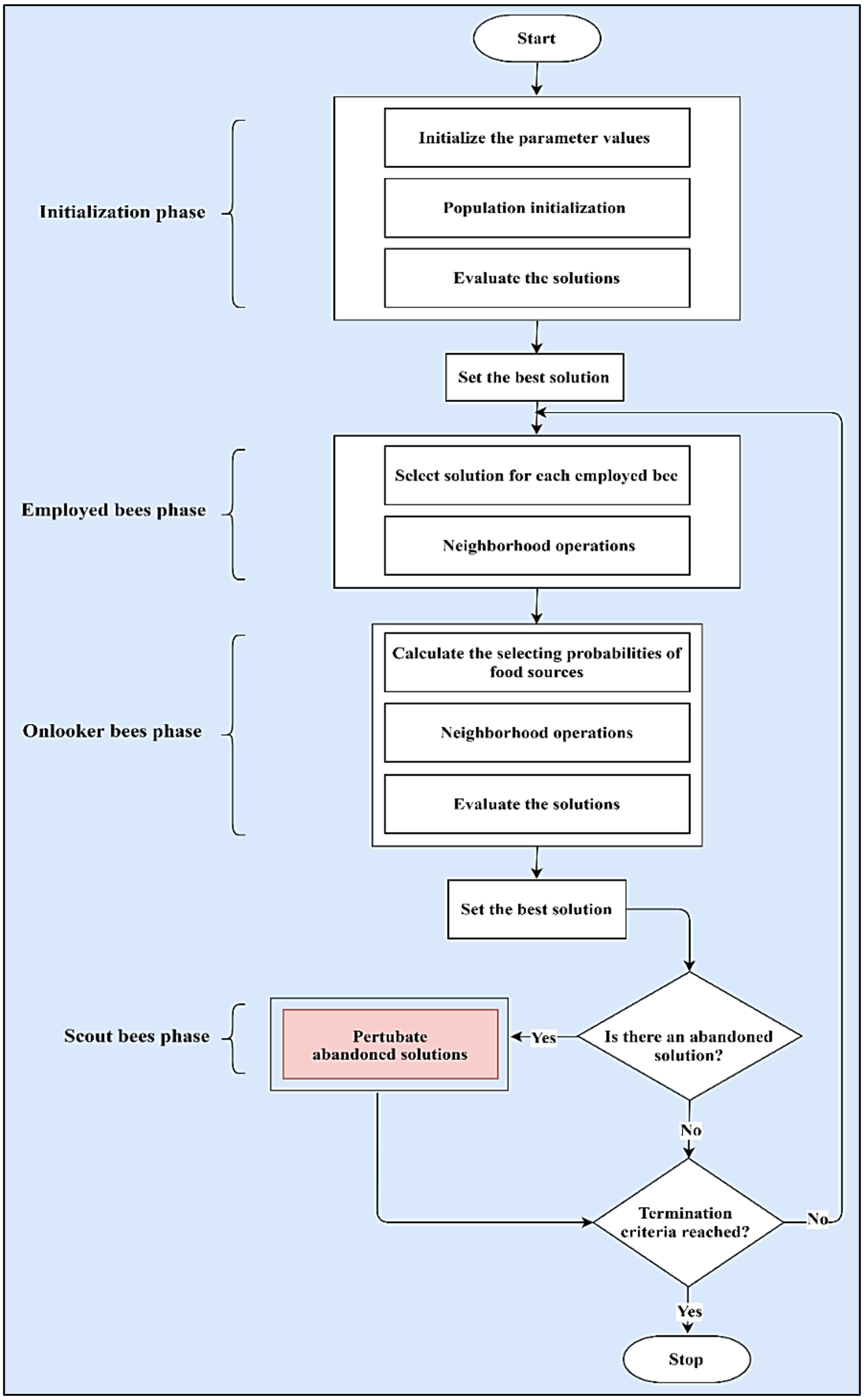

Figure 2. Flowchart of the P-MABC algorithm for VRPTW

5.1. VRPTW benchmark dataset and parameter setting

Computational experiments are conducted on Solomon's VRPTW benchmark. The benchmark is divided into six classes of problems: R1, R2, C1, C2, RC1, and RC2. There are 100 customers, a single warehouse, capacity, and time window restraints are present for all problems. On the basis of customer distribution, customers are randomly distributed into R1 and R2, customers are clustered in C1 and C2. Customers are mixed for randomly distributed and clustered in RC1 and RC2 as shown in Table 1. The problems of R2, C2, and RC2 have long scheduling horizon, whereas those of R1, C1, and RC1 have short scheduling horizon. Table 1 summarizes the characteristics of VRPTW instances and Table 2 presents the deployed parameter settings [30] for the proposed P-MABC algorithm.

Table 1. Features of solomon's VRPTW benchmark

Features	Dataset					
	R1	R2	C1	C2	RC1	RC2
Instances number	12	11	9	8	8	8
Vehicle capacity	200	1,000	200	700	200	1,000
Service time	10	10	90	90	10	10
Time windows width	Tight	Wide	Tight	Wide	Tight	Wide
Customer distribution	Randomly	Randomly	Clustered	Clustered	Mixed	Mixed

Table 2. Parameter settings for P-MABC algorithm

Parameter	Value
Number of iterations	1,000
Population size = EB=OB	50
Limit	100

5.2. Experimental results

Two sets of experiments are performed in this research to investigate the efficacy of the proposed P-MABC algorithm in solving VRPTW. The first experiment investigates the ability of perturbation operations to improve the performance of MABC by enhancing the exploration task, which leads to increased swarm diversity and obtaining good quality solutions. The results of utilized perturbation operations on P-MABC are compared with those of MABC. The second experiment involved the analysis of the performance of P-MABC in solving the VRPTW compared with those of optimization algorithms.

5.2.1. Comparison between P-MABC and MABC algorithm

P-MABC has been assessed on a set of 56 VRPTW benchmarks to evaluate the P-MABC on VRPTW. This experiment aims to compare P-MABC versus MABC. Experimental results are based on 31 independent runs. Tables 3-5 show the performance comparison results in categories R, C, and RC, respectively.

Table 3. Comparison results for the problem R1 and R2

Instance	MABC			P-MABC		
	NV	TD	Average	NV	TD	Average
R101	20	1643.18	1647.91	20	1642.88	1645.87
R102	18	1480.73	1490.66	18	1472.57	1480.71
R103	14	1240.87	1258.50	14	1209.43	1239.51
R104	12	1047.06	1070.27	11	1002.4	1022.9
R105	16	1369.52	1382.35	15	1360.74	1372.32
R106	13	1271.13	1285.81	13	1237.32	1254.19
R107	12	1129.99	1142.23	11	1073.34	1115.28
R108	11	1004.11	1026.11	10	947.66	975.39
R109	13	1170.50	1211.12	13	1148.75	1168.95
R110	12	1123.36	1145.30	12	1071.61	1100.98
R111	12	1101.59	1129.55	12	1048.11	1090.97
R112	11	1019.84	1026.25	10	966.88	993.90
R201	8	1185.57	1192.87	6	1170.25	1175.34
R202	7	1103.15	1114.87	6	1056.84	1060.02
R203	6	958.94	984.34	5	881.12	893.13
R204	4	818.44	836.49	5	740.53	787.09
R205	6	1020.53	1023.79	4	957.48	979.47
R206	5	960.29	976.45	5	885.74	928.53
R207	5	905.70	930.46	4	809.45	821
R208	4	764.90	789.02	3	728.43	734.73
R209	6	943.16	952.73	5	861.46	877.37
R210	6	1003.91	1015.11	5	982.93	979.07
R211	5	837.66	855.79	4	767.22	773.68

Table 4. Comparison results for the problem C1 and C2

	MABC			P-MABC		
Instance	NV	TD	Average	NV	TD	Average
C101	10	828.94	828.94	10	828.93	828.93
C102	10	828.94	828.94	10	828.93	828.93
C103	10	828.94	840.66	10	828.06	828.93
C104	10	858.90	889.10	10	824.78	837.99
C105	10	828.94	828.94	10	828.93	828.93
C106	10	828.94	828.94	10	828.93	828.93
C107	10	828.94	828.94	10	828.93	828.93
C108	10	828.94	830.85	10	828.93	828.93
C109	10	828.94	836.47	10	828.93	828.93
C201	3	591.56	591.56	3	591.56	603.13
C202	3	591.56	601.78	3	591.56	603.56
C203	3	600.54	616.39	3	591.17	601.60
C204	3	610.01	648.57	3	590.6	602.83
C205	3	588.88	596.10	3	588.87	590.66
C206	3	588.88	601.49	3	588.49	591.49
C207	3	589.58	601.60	3	588.29	592.78
C208	3	591.65	613.47	3	591.32	592.08

Table 5. Comparison results for the problem RC1 and RC2

Instance	MABC			P-MABC		
	NV	TD	Average	NV	TD	Average
RC101	16	1,634.52	1,668.07	16	1,633.28	1,654.88
RC102	15	1,492.89	1,505.94	14	1,480.38	1,498.06
RC103	13	1,334.57	1,360.15	12	1,275.35	1,300.94
RC104	11	1,215.62	1,245.35	10	1,140.21	1,169.73
RC105	15	1,546.43	1,575.46	15	1,520.33	1,540.63
RC106	14	1,423.10	1,443.77	13	1,408.22	1,413.5
RC107	12	1,300.00	1,324.00	12	1,212.83	1,264.51
RC108	12	1,193.68	1,213.67	11	1,135.69	1165
RC201	8	1,308.76	1,320.24	7	1,276.48	1,326.99
RC202	8	1,167.00	1,180.48	6	1,112.53	1,130.85
RC203	6	1,014.79	1,032.77	5	935	948.62
RC204	4	881.88	894.76	4	790.65	812.80
RC205	7	1,210.68	1,232.84	7	1,161.67	1,166.39
RC206	6	1,112.38	1,133.99	5	1,089.14	1,096.61
RC207	7	1,059.62	1,076.47	6	970.97	998.97
RC208	5	882.06	898.45	5	784.56	839.15

In Tables 3-5 the travelled distance (TD) and number of vehicles (NV) produced from P-MABC and MABC algorithms are presented. Table 3 demonstrate that P-MABC enables produce better results than the MABC algorithm in terms of distance from all instances in R1 and R2. The P-MABC algorithm utilizes a smaller number of vehicles than MABC for 14 out of 23 instances (60.86%) from R1 and R2.

Table 4 shows that P-MABC produces better results for 15 out of 17 instances (88.23 %) compared with MABC. The present 15 instances (i.e., C101, C102, C103, C104, C203, C204, C206, C207, C208, C109, C203, C204, C205, C206, C207 and C208) in terms of the TD from problem instances C1 and C2. Both algorithms are at equal when comparison is made on NV in problem sets C1 and C2.

Table 5 shows that the P-MABC produces better results compared with MABC in terms of distance from all 16 instances in RC1 and RC2. Furthermore, the P-MABC algorithm utilizes a smaller NV than the MABC for 10 instances out of 16 instances (62.5%) from RC1 and RC2. Given these results, the consecutive conclusion can be made on the P-MABC has improved the performance by addressing the VRPTW as compared to MABC of 56 instances. P-MABC algorithm yield better results in 54 out of 56 out of 56 instances (96.42%) compared with MABC in terms of distance moreover, P-MABC obtains better results than MABC for 24 out of 56 instances (42.85%) in terms of the number of vehicles.

5.2.2. Comparsion between P-MABC and optimization algorithms

The results achieved by P-MABC are compared with those results provided by various optimization algorithms deployed for the VRPTW in terms of NV and TD. The following algorithms have been utilized: localized genetic algorithm (LGA) [25], M-MOEA/D [27], tabu-ABC [28], and the evolutionary scatter search particle swarm optimization algorithm (ESS-PSO) [26]. The TD and NV results of P-MABC versus optimization algorithms for VRPTW problems (i.e., R1, R2, C1, C2, RC1, and RC2) are listed in Tables 6-8.

Table 6. Comparison results for the problem R1 and R2

Instance	LGA		M-MOEA/D		Tabu-ABC		ESS-PSO		P-MABC	
	NV	TD	NV	TD	NV	TD	NV	TD	NV	TD
R101	20	1,646.9	20	1,644.7	20	1643.18	20	1,642.88	20	1,642.88
R102	18	1,474.28	18	1,473.73	18	1460.26	18	1,472.92	18	1,472.57
R103	15	1,222.68	14	1,213.62	15	1217.39	14	1,213.73	14	1,209.43
R104	11	989.53	11	991.91	11	987.61	11	976.61	11	1,002.4
R105	16	1,382.78	15	1,366.56	15	1363.91	15	1,360.76	15	1,360.74
R106	13	1,250.11	13	1,249.22	13	1247.9	13	1,239.37	13	1,237.32
R107	12	1,083.42	11	1,086.22	12	1087.5	11	1,073.34	11	1,073.34
R108	10	952.44	10	965.52	11	961.85	10	950.59	10	947.66
R109	13	1,160.69	13	1,155.38	13	1152.99	13	1,151.84	13	1,148.75
R110	12	1,080.69	12	1,106.03	12	1091.5	12	1,073.46	12	1,071.61
R111	12	1,057.64	11	1,073.82	12	1067.46	12	1,053.5	12	1,048.11
R112	10	965	10	981.43	10	973.25	10	953.62	10	966.88
R201	9	1,156.29	6	1,185.79	6	1,174.69	9	1,148.48	6	1,170.25
R202	8	1,042.25	5	1,049.72	5	1,046.1	7	1,049.74	6	1,056.84
R203	6	877.29	5	889.36	5	884.02	5	900.08	5	881.12
R204	4	736.52	5	743.29	4	750.4	4	772.33	5	740.53
R205	6	960.35	5	954.48	5	960.75	6	970.89	4	957.48
R206	6	894.19	4	887.9	4	900.97	5	898.91	5	885.74
R207	4	800.79	4	809.51	4	809.72	3	814.78	4	809.45
R208	3	706.86	3	711.59	5	723.14	3	723.61	3	728.43
R209	5	860.63	4	867.47	5	863.12	6	879.53	5	861.46
R210	5	948.82	5	920.06	5	927.54	7	932.89	5	982.93
R211	5	762.23	4	767.1	4	763.22	4	808.56	4	767.22

Table 7. Comparison results for the problem C1 and C2

Instance	LGA		M-MOEA/D		Tabu-ABC		ESS-PSO		P-MABC	
	NV	TD	NV	TD	NV	TD	NV	TD	NV	TD
C101	10	828.94	10	828.94	10	828.94	10	828.94	10	828.93
C102	10	828.94	10	828.94	10	828.94	10	828.94	10	828.93
C103	10	828.06	10	828.06	10	828.06	10	828.06	10	828.06
C104	10	824.87	10	824.87	10	824.87	10	824.78	10	824.78
C105	10	828.94	10	828.94	10	828.94	10	824.94	10	828.93
C106	10	828.94	10	828.94	10	828.94	10	828.94	10	828.93
C107	10	828.94	10	828.94	10	828.94	10	828.94	10	828.93
C108	10	828.94	10	828.94	10	828.94	10	828.94	10	828.93
C109	10	828.94	10	828.94	10	828.94	10	828.94	10	828.93
C201	3	591.56	3	591.56	3	591.56	3	591.56	3	591.56
C202	3	591.56	3	591.56	3	591.56	3	591.56	3	591.56
C203	3	591.17	3	591.17	3	591.17	3	591.17	3	591.17
C204	3	590.6	3	590.6	3	594.89	3	590.6	3	590.6
C205	3	588.88	3	588.88	3	588.88	3	588.88	3	588.87
C206	3	588.49	3	588.49	3	588.49	3	588.49	3	588.49
C207	3	588.29	3	588.29	3	588.29	3	588.29	3	588.29
C208	3	588.32	3	588.32	3	588.32	3	588.32	3	588.32

Table 6 represent the P-MABC algorithm results versus optimization algorithms in problem instances R1 and R2. The P-MABC algorithm gives better results for 12 out of 23 instances (52.17%) than LGA, 16 out of 23 instances (69.56%) than M-MOEA/D, 15 out of 23 instances (65.21%) than ESS-PSO, and 17 out of 23 instances (73.91%) than tabu-ABC. At the same NV, the P-MABC algorithm gives better results for nine instances and competitive results for 13 instances with the same NV as compared to the LGA, the P-MABC gives better results for one instance and competitive results for 18 instances with the same NV as compared to the M-MOEA/D, the P-MABC gives better results for five instances and competitive results for 15 instances with same NV as compared to the tabu-ABC, and the P-MABC gives better results for five instances and competitive results for 16 instances with same NV as compared to the ESS-PSO. Table 7 illustrates that P-MABC has accomplished better results in terms of distance than LGA, M-MOEA/D, Tabu-ABC, and ESS-PSO in 8 (i.e., C101, C101, C105, C105, C107, C108, C109, and C205) out of 17 instances (47.05%) from problem instances C1 and C2. All algorithms are equal in problem sets C1 and C2 to compare the NV amongst these algorithms.

The obtained results by P-MABC versus optimization algorithms in problem sets RC1 and RC2 are shown in Table 8. Comparison with LGA verifies that P-MABC produces best distance in nine out of 16 instances (56.25%). P-MABC beats M-MOEA/D in 12 out of 16 instances from (RC1 and RC2). P-MABC is better for 13 out of 16 instances (81.25%) than Tabu-ABC. Also, the P-MABC algorithm obtained best results than the ESS-PSO in two out of 16 instances from RC1 and RC2.

Correspondingly, the P-MABC algorithm gives better results for nine instances and competitive results for seven instances with the same NV as compared to the LGA. P-MABC gives better results for one instance and competitive results for 10 instances with the same NV as compared to the M-MOEA/D. P-MABC gives better results for three instances and competitive results for 12 instances with same NV as compared to the Tabu-ABC and ESS-PSO.

The outcomes of Tables 6-8 can be summarized through the following conclusion made on the P-MABC performance versus the state-of-the-art algorithms in VRPTW instances. On the basis of the 56 instances of VRPTW, the results obtained by the P-MABC algorithm yield good solutions compared with optimization algorithms. The P-MABC algorithm is better than the LGA by 51.78%, M-MOEA/D and Tabu-ABC by 64.28%, and ESS-PSO by 48.21% in terms of TD. Furthermore, the P-MABC algorithm continues to be efficient in terms of overall NV. P-MABC algorithm is better than the LGA by 25.07%, M-MOEA/D by 3.57%, Tabu-ABC and ESS-PSO by 14.28%, in terms of NV.

Table 8. Comparison results for the problem RC1 and RC2

Instance	LGA		M-MOEA/D		Tabu-ABC		ESS-PSO		P-MABC	
	NV	TD	NV	TD	NV	TD	NV	TD	NV	TD
RC101	16	1,660.55	16	1,646.65	16	1,646.17	16	1,639.75	16	1,633.28
RC102	15	1,494.92	15	1,484.48	14	1,481.61	14	1,461.33	14	1,480.38
RC103	12	1,276.05	11	1,274.85	12	1,280.76	12	1,277.55	12	1,275.35
RC104	10	1,151.63	10	1,145.79	11	1,162.03	10	1,138.13	10	1,140.21
RC105	16	1,556.21	15	1,528.61	16	1,545.3	15	1,519.46	15	1,520.33
RC106	14	1,402.25	13	1,399.17	14	1,401.17	13	1,378.62	13	1,408.22
RC107	12	1,212.83	12	1,235.54	12	1,235.28	12	1,212.83	12	1,212.83
RC108	11	1,133.25	11	1,138.95	11	1,136.35	11	1,118.57	11	1,135.69
RC201	10	1,281.63	7	1,289.94	7	1,271.78	9	1,265.56	7	1,276.48
RC202	8	1,103.47	5	1,118.66	6	1,116.21	8	1,096.53	6	1,112.53
RC203	6	942	5	940.55	5	941.81	5	926.82	5	935
RC204	4	796.12	4	792.98	4	801.87	4	786.38	4	790.65
RC205	8	1,168.89	6	1,187.48	7	1,165.82	7	1,157.55	7	1,161.67
RC206	7	1,060.52	5	1,089.14	5	1,072.85	6	1,057.83	5	1,089.14
RC207	7	970.97	5	987.88	5	977.11	6	966.37	6	970.97
RC208	5	782.7	4	807.83	5	792.33	4	779.31	5	784.56

The results obtained in this research demonstrate that the proposed P-MABC is an effective solution algorithm for the VRPTW. This finding can be attributed to the applicability of the proposed P-MABC that has different perturbation operators in the abandoned solutions for different problem instances. By utilizing different operators during the search, the P-MABC can deal with various problem instances as well as the changes that might happen during the optimization of a solution. Based on these results, the P-MABC raised in section 5.2.2 is accepted and proved to be true.

Table 9 and Figure 3 illustrate the statistical test of the performance of P-MABC versus other optimization algorithms based on the nonparametric Friedman test with Holm's post hoc test. This test aims to find the dominant P-MABC among 56 datasets. The average TD, and the average NV are included. As Table 9, in all cases, the lowest rank indicates good algorithm performance. Consequently, the P-MABC achieves the best average rank in TD and the second average rank in the NV.

Table 9. Overall results on the average perfomance

	Algorithm					
	MABC	LGA	M-MOEA/D	Tabu-ABC	ESS-PSO	P-MABC
Distance	5.43	3.31	3.68	3.59	2.69	2.26
Number of vehicles	4.14	3.99	2.79	3.54	3.43	3.05

Figure 3 demonstrates the result of the average TD rank versus the NV rank. The P-MABC acquires the best TD and the second in the NV. Under these circumstances, the P-MABC dominates the perturbation operators versus optimization algorithms in two assessment criteria. This result is due to the enhancement process in the exploration phase accomplished by using five types of perturbation operators to help the MABC. Perturbation operators utilize swarm diversity and to find promising areas that have good-quality solutions.

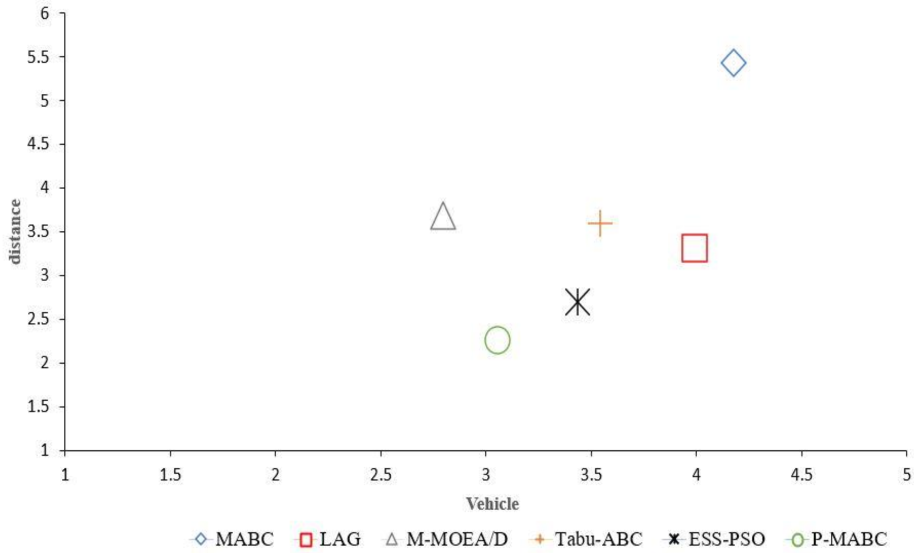

Figure 3. Average rank test of the P-MABC versus optimization algorithms

6. CONCLUSION

In this research, the MABC exploration process and the quality of generated solutions by SB have been enhanced by five perturbation operations adapted for the MABC called the P-MABC algorithm. This enhancement made the P-MABC has a good exploration process to explore the search space. Experimental results have proven that the P-MABC improves the solutions through SB when using neighborhood operation as a perturbation search to improve the solution quality and encourage search diversification, discover the less-crowded region in the existing archive, and achieve good-quality solutions. The overall comparison indicates that the P-MABC algorithm outperforms the MABC algorithm alone. In addition, P-MABC can achieve the best results in comparison with other optimization algorithms by reducing the cost of TD and the NV, which are the main objectives of the VRPTW. The proposed P-MABC will be used in VRPTW applications, such as material handling systems and bank delivery, to explore its performance further.

REFERENCES

[1] A. O. Adewumi and O. J. Adeleke, "A survey of recent advances in vehicle routing problems," *International Journal of System Assurance Engineering and Management*, vol. 9, no. 1, pp. 155–172, Feb. 2018, doi: 10.1007/s13198-016-0493-4.
[2] M. Blocho, "Exact algorithms for solving rich vehicle routing problems," in *Smart Delivery Systems*, Elsevier, 2020, pp. 93–99.
[3] R. Elshaer and H. Awad, "A taxonomic review of metaheuristic algorithms for solving the vehicle routing problem and its variants," *Computers & Industrial Engineering*, vol. 140, p. 106242, Feb. 2020, doi: 10.1016/j.cie.2019.106242.
[4] A. S. Nejad and G. Fazekas, "Solving a traveling salesman problem using meta-heuristics," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 11, no. 1, p. 41, Mar. 2022, doi: 10.11591/ijai.v11.i1.pp41-49.
[5] A. G. Ioannou, M. Kritikos, and G. Prastacos, "A greedy look-ahead heuristic for the vehicle routing problem with time windows," *J. Oper. Res. Soc.*, vol. 52, no. 5, pp. 523–537, 2001.
[6] Loo Hay Lee, Kay Chen Tan, Ke Ou, and Yoong Han Chew, "Vehicle capacity planning system: A case study on vehicle routing problem with time windows," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 33, no. 2, pp. 169–178, Mar. 2003, doi: 10.1109/TSMCA.2002.806498.
[7] H. K. Chen, C. F. Hsueh, and M.-S. Chang, "Production scheduling and vehicle routing with time windows for perishable food products," *Computers & Operations Research*, vol. 36, no. 7, pp. 2311–2319, Jul. 2009, doi: 10.1016/j.cor.2008.09.010.
[8] A. Boonkleaw, N. Suthikarnnarunai, and R. Srinon, "Strategic planning for newspaper delivery problem using vehicle routing algorithm with time window (VRPTW)," *Eng. Lett*, vol. 18, no. 2, 2010.
[9] A. Benantar, R. Ouafi, and J. Boukachour, "A petrol station replenishment problem: new variant and formulation," *Logistics Research*, vol. 9, no. 1, p. 6, Dec. 2016, doi: 10.1007/s12159-016-0133-z.
[10] M. M. Solomon, "Algorithms for the vehicle routing and scheduling problems with time window constraints," *Operations Research*, vol. 35, no. 2, pp. 254–265, Apr. 1987, doi: 10.1287/opre.35.2.254.
[11] K. Hussain, M. N. Mohd Salleh, S. Cheng, and Y. Shi, "Metaheuristic research: a comprehensive survey," *Artificial Intelligence Review*, vol. 52, no. 4, pp. 2191–2233, Dec. 2019, doi: 10.1007/s10462-017-9605-z.
[12] H. N. K. Al-Behadili, K. R. Ku-Mahamud, and R. Sagban, "Genetic-based Pruning technique for ant-miner classification algorithm," *International Journal on Advanced Science, Engineering and Information Technology*, vol. 11, no. 1, p. 304, Feb. 2021, doi: 10.18517/ijaseit.11.1.10826.
[13] A. M. Jabbar, K. R. Ku-Mahamud, and R. Sagban, "Improved self-adaptive ACS algorithm to determine the optimal number of clusters," *International Journal on Advanced Science, Engineering and Information Technology*, vol. 11, no. 3, p. 1092, Jun. 2021, doi: 10.18517/ijaseit.11.3.11723.
[14] H. F. Almazini, S. Mortada, H. F. Abbas Al-Mazini, H. N. Khraibet AL-Behadili, and J. Alkenani, "Improved discrete plant propagation algorithm for solving the traveling salesman problem," *IAES International Journal of Artificial Intelligence (IJ-AI)*,

vol. 11, no. 1, p. 13, Mar. 2022, doi: 10.11591/ijai.v11.i1.pp13-22.
[15] H. Almazini and K. Ku-Mahamud, "Grey wolf optimization parameter control for feature selection in anomaly detection," *International Journal of Intelligent Engineering and Systems*, vol. 14, no. 2, pp. 474–483, Apr. 2021, doi: 10.22266/ijies2021.0430.43.
[16] H. Almazini and K. Ku-Mahamud, "Adaptive technique for feature selection in modified graph clustering-based ant colony optimization," *International Journal of Intelligent Engineering and Systems*, vol. 14, no. 3, pp. 332–345, Jun. 2021, doi: 10.22266/ijies2021.0630.28.
[17] J. Dréo, A. Pétrowski, P. Siarry, and E. Taillard, *Metaheuristics for Hard Optimization*. Berlin/Heidelberg: Springer-Verlag, 2006.
[18] H. N. K. Al-Behadili, "Improved firefly algorithm with variable neighborhood search for data clustering," *Baghdad Science Journal*, vol. 19, no. 2, Apr. 2022, doi: 10.21123/bsj.2022.19.2.0409.
[19] E. G. Talbi, "Metaheuristics: From Design to Implementation," *Metaheuristics: From Design to Implementation*, 2009, doi: 10.1002/9780470496916.
[20] H. F. Almazini, "at algorithm optimisation technique for feature selection on different dimension of datasets," *Iraqi Journal of Intelligent Computing and Informatics*, vol. 1, no. 1, pp. 11–21, 2022.
[21] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm," *Journal of Global Optimization*, vol. 39, no. 3, pp. 459–471, Oct. 2007, doi: 10.1007/s10898-007-9149-x.
[22] A. M. Jabbar, "Rule induction with iterated local search," *International Journal of Intelligent Engineering and Systems*, vol. 14, no. 4, pp. 289–298, Aug. 2021, doi: 10.22266/ijies2021.0831.26.
[23] A. C. Kamil, J. R. Krebs, and H. R. Pulliam, *Foraging behavior*. Springer Science and Business Media, 2012.
[24] Z. Ursani, D. Essam, D. Cornforth, and R. Stocker, "Localized genetic algorithm for vehicle routing problem with time windows," *Applied Soft Computing*, vol. 11, no. 8, pp. 5375–5390, Dec. 2011, doi: 10.1016/j.asoc.2011.05.021.
[25] J. Zhang, F. Yang, and X. Weng, "An evolutionary scatter search particle swarm optimization algorithm for the vehicle routing problem with time windows," *IEEE Access*, vol. 6, pp. 63468–63485, 2018, doi: 10.1109/ACCESS.2018.2877767.
[26] Y. Qi, Z. Hou, H. Li, J. Huang, and X. Li, "A decomposition based memetic algorithm for multi-objective vehicle routing problem with time windows," *Computers & Operations Research*, vol. 62, pp. 61–77, Oct. 2015, doi: 10.1016/j.cor.2015.04.009.
[27] D. Zhang, S. Cai, F. Ye, Y.-W. Si, and T. T. Nguyen, "A hybrid algorithm for a vehicle routing problem with realistic constraints," *Information Sciences*, vol. 394–395, pp. 167–182, Jul. 2017, doi: 10.1016/j.ins.2017.02.028.
[28] S. Mortada and Y. Yusof, "A neighbourhood search for artificial bee colony in vehicle routing problem with time windows," *International Journal of Intelligent Engineering and Systems*, vol. 14, no. 3, pp. 255–266, Jun. 2021, doi: 10.22266/ijies2021.0630.22.
[29] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," *Tech. report-tr06, Erciyes Univ. Eng. Fac. Comput. Eng. Dep.*, vol. 200, pp. 1–10, 2005.
[30] M. Alzaqebah, S. Abdullah, and S. Jawarneh, "Modified artificial bee colony for the vehicle routing problems with time windows," *SpringerPlus*, vol. 5, no. 1, p. 1298, Dec. 2016, doi: 10.1186/s40064-016-2940-8.

BIOGRAPHIES OF AUTHORS

Salah Mortada is a Doctor of Philosophy at Shatt Al-Arab University College, Basra, Iraq. He holds a master's degree in information technology in the area (Usability and User Experience). His current research interests include artificial intelligence, evolutionary computation, meta-heuristic algorithms, swarm intelligence, and their applications in the design and optimization of intelligent transportation management such as VRPs. He can be contacted at email: salah.m.shaheen@sa-uc.edu.iq.

Yuhanis Yusof is an associate professor at Universiti Utara Malaysia (UUM). She has a Ph.D. in ComputerScience from Cardiff University, United Kingdom. She also holds a MSc. degree in Computer Science from University Sains Malaysia (USM), and a Bachelor of Information Technology from UUM. Her research interest is broadly in data analytics and management for large-scale computing. This includes data mining (discovering patterns of interest from data), information retrieval, and optimization. Currently, she is involved in several projects relating to machine Learning and Swarm Intelligence (e.g., Artificial Bee Colony, Firefly algorithm, Grey Wolf optimizer, Cuckoo Search, and Bat algorithm). She can be contacted at email: yuhanis@uum.edu.my.