A Hybrid Max-Min Ant System Algorithm for Electric Capacitated Vehicle Routing Problem

Yan-e Hou, Congran Wang, Chunyang Zhang, Lanxue Dang, and Chunjing Xiao

Abstract—**Electric vehicles have become the first choice for products delivery in logistics, due to their environmental friendliness and low cost compared to the traditional fuel vehicles. This paper deals with an electric capacitated vehicle routing problem with variable energy consumption rate (E-CVRP), which aims to minimize the total cost including vehicle fixed cost, traveled distance cost and energy cost. First, a mixed integer linear programming model is constructed based on the fact that the energy consumption is affected by the vehicle load. Second, a hybrid Max-Min Ant System (HMMAS) algorithm is proposed to solve this problem. Three neighborhood structures are introduced and executed in a Variable Neighborhood Descent (VND) algorithm to improve the current solution. In addition, an effective charging station adjustment strategy is adopted to further improve the solution obtained by the local search process. Finally, some experiments have been conducted on public instances, and the results demonstrate the effectiveness and stability of the algorithm.**

Index Terms—**electric vehicles, vehicle routing problem, max-min ant system, variable neighborhood descent, charging station adjust.**

I. Introduction

In recent years, greenhouse gas emissions, particularly CO_2, have had a major impact on the worlds ecosystem and social-economic development. Reducing carbon emissions is crucial to protect the Earths ecological environment and the long-term human development. The transportation sector is the second largest sector of the worlds total carbon emissions[1]. To achieve the low-carbon goal, the application of electric vehicles (EVs) is an effective way to reduce CO_2 emissions in the transportation domain. Electric vehicles have gradually replaced the traditional fuel vehicles in public transportation and logistics fields, as they have some advantages of environmental friendliness, low cost and energy consumption. It is generally agreed that reasonable planning of electric vehicle routes can not only reduce carbon emissions, but also improve vehicle utilization and reduce operating costs.

Manuscript received September 15, 2023; revised December 24, 2023.This research has been supported by Key Research Projects of Higher Education Institutions of Henan Province, China (23A520014).

Yan-e Hou is an associate professor of Henan Key Laboratory of Big Data Analysis and Processing, Henan University, Kaifeng, 475004 China (e-mail: houyane@henu.edu.cn).

Congran Wang is a graduate student of College of Computer and Information Engineering, Henan University, Kaifeng, 475004 China (e-mail: wangcongran@henu.edu.cn).

Chunyang Zhang is a graduate student of College of Computer and Information Engineering, Henan University, Kaifeng, 475004 China (e-mail: cyzh@vip.henu.edu.cn).

Lanxue Dang is a professor of College of Computer and Information Engineering, Henan University, Kaifeng, 475004 China (e-mail: danglx@vip.henu.edu.cn).

Chunjing Xiao is an associate professor of School of Computer and Information Engineering, Henan University, Kaifeng, 475004 China (Corresponding author, e-mail: Chunjingxiao@gmail.com).

The Electric Vehicle Routing Problem (EVRP) is an important variant of the classical Vehicle Routing Problem (VRP). Unlike traditional VRP, EVRP has battery quantity constraint except for the constraints commonly used in VRP such as vehicle capacity, time windows, etc. In EVRP, each battery quantity of electric vehicle decreases as the vehicle travels. Therefore, when there is not enough battery quantity to serve the subsequent customers, the electric vehicle has to go to a charging station to recharge. When planning the routes of electric vehicles, it is necessary to consider more factors such as the vehicle battery and the location of the charging station, etc. For this reason, the EVRP problem is more complex than the classical VRP problem.

As an important variant of VRP, there are a number of research results in the area of EVRP. Conrad and Figliozzi [2] first studied the recharging vehicle routing problem and took into account the vehicle recharging while driving. Green vehicle routing problem was proposed by [3], which inspired the subsequent studies. Schneider et al. [4] designed a hybrid variable neighborhood search and tabu search algorithm to solve the EVRP with time window (EVRPTW). Goeke and Schneider [5] presented an adaptive large neighborhood search (ALNS) method for mixed fleets EVRP, where the energy consumption is calculated by a realistic energy consumption function. Lin et al. [6] considered the impact of vehicle load on energy consumption for EVRP. Keskin and Catay [7] proposed a partial recharging strategy and designed an ALNS algorithm to solve EVRPTW with partial recharging. Montoya et al. [8] used non-linear charging functions to calculate battery consumption, and then developed a hybrid iterated local search and hill climbing algorithm to solve the EVRP with non-linear charging function. Keskin et al. [9] implemented an adaptive large neighborhood search algorithm to solve EVRP problem, which considered vehicle queuing at charging stations. The recent survey of EVRP can be found in [10],[11].

These literatures have prompted the development of EVRP to some extent. However, it still needs to further exploit the research of EVRP. Many researchers have only focused on minimizing the total travel distance, which does not reflect the real operation cost. In addition, the existing EVRP model rarely considers the impact of vehicle load on battery consumption. When the load on the electric vehicle is very big, it requires more battery consumption. Reference [6] indicates that the effect of vehicle load on energy consumption should not be ignored. The energy consumption rate is linearly related to the vehicle load. Therefore, it is necessary to study the EVRP with total cost including vehicle fixed cost, operation cost and energy consumption cost, and the calculation of energy consumption should consider the impact of the vehicle load, that is electric capacitated vehicle routing problem (E-CVRP). Like VRP, E-CVRP is also a

NP-hard problem, so developing effective algorithms for E-CVRP remains a challenge.

Ant Colony Optimization (ACO) algorithm was proposed by [12] in 1991. ACO algorithms include several different variants, all of which are inspired by the foraging behavior of some ant species. The max-min ant system (MMAS) [13] is one of the most successful variants of all ACO algorithms. It is characterized by the fact that only the best ant updates the pheromone trails, and that the value of the pheromone is bounded. MMAS has been successfully applied in solving VRP and EVRP [14],[15],[16],[17],[18],[19].

In view of, this paper deals with the E-CVRP problem with the total operating cost, which is made up of vehicle fixed cost, total distance traveled cost, and total energy consumption cost. Then, a hybrid MMAS algorithm, called HMMAS, combined with VND is proposed to solve the model. In the process of VND, three neighborhood operators are used to find a better solution. Additionally, we developed a charging station adjustment strategy to adjust the locations of the charging stations to enhance the quality of solutions. The experimental results on some public instances reveal the proposed algorithm is effective and stable compared to existing EVRP algorithms.

The remaining structure of this paper is as follows. Section II describes and models the addressed problem. The proposed HMMAS algorithm in this paper is introduced in Section III. Section IV give the experimental results and analysis. Finally, we draw a remark and give the research direction in Section V.

II. Problem definition and formulation

A. Problem Definition

The E-CVRP is described as follows. A depot provides delivery services to a set of customers with known demand. The deliveries are performed by a homogeneous fleet of electric vehicles with load capacity and battery capacity. All vehicle routes start and end at the depot. While the vehicle travels, the battery quantity decreases proportionally to its load and distance traveled. Therefore, the EV may need to visit a charging station to continue its route. Each vehicle departs from the depot or recharging stations fully charged. Travel speeds are assumed to be constant along each arc. The E-CVRP aims to find an optimal set of routes to minimize operating costs, subject to several constraints.

B. Model Formulation

The E-CVRP is defined on a directed graph $G = (V, E)$. Assume that $N = \{1, 2, \ldots, n\}$ denotes the set of customers, 0 represents the depot, and F is the set of charging stations. F' denotes the set of β_i copies of each charging station $i \in F$ (i.e., $|F'| = \sum_{i \in F} \beta_i$). The set $|F'|$ allows each charging station to be visited multiple times [3]. β_i is set to $2|N|$ since, in the worst case, each EV needs to visit the charging station once before and after serving each customer [20]. So, the set of all nodes is denoted as $V = \{0\} \cup N \cup F'$. The set of arcs is denoted by $E = \{(i,j), \forall i,j \in V, i \neq j\}$. Each node $i \in V$ has a non-negative demand q_i, with $q_i = 0$ for $i \notin N$. Each EV has a load capacity of C and a battery capacity of Q. Each arc between node i and node j is assigned a distance d_{ij} and a load-dependent energy consumption $h_i d_{ij}$. Due to the uncertainty load carried on arc (i,j), h_i is a variable energy consumption rate [21] is defined as:

$$h_i = \left(r + \frac{u_i}{C}\right) \tag{1}$$

Where r is a constant and denotes each arc's empty-vehicle energy consumption rate, variable u_i refers to the load of the vehicle carried on arc (i,j). Variable y_i refers to the remaining battery when the vehicle arrives at node i. The binary decision variable x_{ij} determines whether the vehicle travels on arc (i,j). If so, $x_{ij} = 1$, otherwise $x_{ij} = 0$. Each vehicle departs from the depot fully loaded and fully charged.

The E-CVRP is to find optimal delivery routes for a fleet of EVs, considering battery capacity and load capacity constraints. The total cost of routes consists of three parts: vehicle fixed cost Z_1, distance traveled cost Z_2 and energy cost Z_3. Vehicle fixed cost is a function of the number of vehicles being used. It comprehensively considers the driver's wages, vehicle purchase cost, vehicle insurance cost, and so on. Distance traveled cost depends on the total route duration of all EVs. Energy costs are related to electricity consumed and electricity price. They are defined in Equations (2), (3) and (4), where c_1, c_2 and c_3 are the unit cost coefficients.

$$Z_1 = c_1 \sum_{i \in V} x_{0i} \tag{2}$$

$$Z_2 = c_2 \sum_{i,j \in V, i \neq j} d_{ij} x_{ij} \tag{3}$$

$$Z_3 = c_3 \sum_{i,j \in V, i \neq j} h_i d_{ij} x_{ij} \tag{4}$$

The model of E-CVRP is described as follows.
Minimize

$$Z = Z_1 + Z_2 + Z_3 \tag{5}$$

Subject to:

$$\sum_{i \in V, i \neq j} x_{ij} - \sum_{j \in V, i \neq j} x_{ji}, j \in V \tag{6}$$

$$\sum_{i \in N, i \neq j} x_{ij} = 1, \forall j \in V \tag{7}$$

$$\sum_{i \in V, i \neq j} x_{ij} \leq 1, \forall j \in F' \tag{8}$$

$$u_0 = C \tag{9}$$

$$0 \leq u_j \leq u_i - q_j x_{ij} + C(1 - x_{ij}), \forall i,j \in V, i \neq j \tag{10}$$

$$y_0 = Q \tag{11}$$

$$0 \leq y_j \leq y_i - h_{ij} d_{ij} + Q(1 - x_{ij}), \forall i \in V, \forall j \in N \tag{12}$$

$$y_j = Q - h_{ij} d_{ij} x_{ij}, \forall i \in F', \forall j \in 0 \cup N \tag{13}$$

$$x_{ij} \in \{0, 1\}, \forall i \in V, \forall j \in V, i \neq j \tag{14}$$

The objective function (5) minimizes the sum of vehicle fixed cost, distance traveled cost and energy cost. Constraint (6) handles the connectivity of all nodes. Constraint (7) guarantees that each customer is visited exactly once, whereas constraint (8) ensures that each charging station is visited at most once. Constraint (9) enforces the EV is fully loaded from the depot. Constraint (10) tracks the load of the

Algorithm 1 HMMAS Framework

Input: instance data, maximum iterations number I, population size M, neighborhood structures NBS, parameters α, β, $T_{\max}$, $T_{\min}$

Output: the best solution S_{gb}

1: Generate an initial solution S_0 using sweep algorithm
2: Initialize heuristic information matrix; Initialize pheromone trails matrix and the bound values of the pheromone $[\tau_{\min}, \tau_{\max}]$ by S_0; set $T = T_{\max}$
3: set iteration variable $i \leftarrow 1$
4: **repeat**
5: **for** $k = 1, \ldots, M$ **do**
6: construct k-th ant solution
7: **end for**
8: find the best solution S_{ib} in M ant solutions
9: **if** $(f(S_{ib}) - f(S_{gb}))/f(S_{ib}) \leq T$ **then**
10: $S_{ib} \leftarrow VND(S_{ib}, NBS)$
11: **end if**
12: **if** $f(S_{ib}) \leq f(S_{gb})$ **then**
13: $S_{gb} \leftarrow S_{ib}$
14: **end if**
15: update pheromone trails by the method defined by equations (19) and (20)
16: $T \leftarrow T - (T_{\max} - T_{\min})/I$
17: $i \leftarrow i + 1$
18: **until** $i > I$
19: **return** S_{gb}

vehicle and the remaining vehicle load between 0 and the maximum load capacity C. Constraint (11) ensures the EV is fully charged from the depot. Constraint (12) tracks the remaining battery level at each vertex. Constraint (13) defines the remaining battery level when the EV departs from the station. Finally, constraint (14) illustrates the domains of the decision variable.

III. The proposed algorithm

A. Overall Description of HMMAS

MMAS [13] is one of the most effective methods for solving EVRP. It prevented premature convergence of the algorithm by limiting the value of the pheromone trail to the range $[\tau_{\max}, \tau_{\max}]$. However, the results of the standard MMAS fluctuate between the global solution and the local solution when solving medium and large-scale problems. The proposed in this paper applies the VND algorithm to optimize the iterative solution for a given threshold at each iteration. The procedures of HMMAS are shown in Algorithm 1.

In Algorithm1, a feasible initial solution S_0 is constructed for E-CVRP. Then, we initialize the pheromone trail, heuristic information matrix and the pheromone extend values $\tau_{\max}$ and $\tau_{\min}$. The main process of MMAS is in lines 4∼18. Firstly, we constructed M ants of the colony in lines 5∼7. Then, the best ant is obtained in line 8. After that, a new solution S_{ib} is improved if the gap between the latest and the current global optimal solution S_{gb} is smaller than a threshold in lines 9∼11. After finishing an iteration, we update the pheromone trail by section C. Finally, the HMMAS returns a global best solution.

B. Solutions Construction

Each ant constructs a feasible solution by using heuristic information and pheromones. The process of constructing an ant solution is presented in Algorithm 2.

Initially, all ants start from the depot fully loaded and fully charged. Then, each ant iteratively selects the next node until all customers have been visited. Ant *k-th* preference customer as the next node. When the ant *k-th* visits the next node and violates the load constraint, the ant returns to the depot. When the ant *k-th* visits the next node, which violates the battery constraint, the ant detours to the station to charge.

The formula of the *k-th* ant selects customer j from node i is defined as follows:

$$j = \begin{cases} argmax(\tau_{ij}^{\alpha} \cdot \eta_{ij}^{\beta}), & q \leq q_0 \\ Z, & q > q_0 \end{cases} \tag{15}$$

In which τ_{ij} is the pheromone value on arc (i, j), and η_{ij} ($\eta_{ij} = 1/d_{ij}$]) is the heuristic information of arc (i, j). α and β are the relative importance of τ_{ij} and η_{ij}, respectively. q is a random variable uniformly distributed in the range $[0, 1]$, q_0 is the threshold of q. $argmax$ is a function to find the node j that can generate a maximum value of $\tau_{ij}^{\alpha} \cdot \eta_{ij}^{\beta}$. Z is the randomly selected method based on the Roulette Wheel according to the probability in formula (16).

$$P_{ij}^{k} = \begin{cases} \frac{\tau_{ij}^{\alpha} \cdot \eta_{ij}^{\beta}}{\sum_{i \in V} \tau_{ij}^{\alpha} \cdot \eta_{ij}^{\beta}}, & j \in J_k \\ 0, & j \notin J_k \end{cases} \tag{16}$$

Where P_{ij}^{k} donates probability that k-th ant from node i selects candidate customer j. J_k represents the unvisited customers of k-th ant. To avoid directly visiting customer j violating constraints, we forecast the state of the route. If customer j violates load constraint, the vehicle returns to the depot. If customer j violates battery constraint, the vehicle detours to the station s. The station s can be expressed by Equation (17).

$$s = argmin\{d_{is} + d_{sj}\} \tag{17}$$

C. Pheromone updating

The proposed algorithm uses the pheromone trails update strategy of MMAS [13]. At the beginning, all the pheromone trails are initialized as follows:

$$\tau_0 = \frac{1}{\rho C_0} \tag{18}$$

Where, ρ ($0 < \rho \leq 1$) indicates the pheromone evaporation factor. C_0 is the cost of a solution generated by the sweep heuristic. After all ants have been constructed to obtain an E-CVRP solution, the pheromone intensity on each edge will be updated based on the best solution. All the pheromone trails are decreasing by a constant factor ρ. After evaporation, the best ant deposits pheromone on the traversed arcs. The updating rules of pheromone trails are shown in Equations (19) and (20).

$$\tau_{\mathrm{ij}}(\mathrm{t}+1) = [(1-\rho) \cdot \tau_{\mathrm{ij}}(t) + \rho \Delta \tau_{\mathrm{ij}}(t)]_{\tau_{\min}}^{\tau_{\max}} \tag{19}$$

Algorithm 2 Solution Construction

Input: instance data, pheromone trails, heuristic information matrix, parameters α, β.
Output: the ant solution S.

Initialize customer unvisited list $J_k \leftarrow \{1, 2, \dots, n\}$
$S \leftarrow \emptyset$, $r \leftarrow \{0\}$
repeat
  select candidate node j from J_k by formulas (15), (16)
  if j violates load constraint **then**
    $j \leftarrow 0$
  end if
  if j violates battery constraint **then**
    $j \leftarrow$ select station by formula (17)
  end if
  $r \leftarrow r \cup \{j\}$
  if $j = 0$ **then**
    $S \leftarrow S \cup \{r\}$
    $r \leftarrow \{0\}$
  else if $1 \leq j \leq n$ **then**
    $J_k \leftarrow J_k - \{j\}$
  end if
until J_k is empty
if J_k is $\emptyset$ **then**
  $r \leftarrow r \cup \{0\}$
  $S \leftarrow S \cup \{r\}$
end if
return S

$$\Delta\tau_{ij}(t) = \begin{cases} \frac{1}{f(S_{ib})}, & (i,j) \in S_{ib} \\ 0, & otherwise \end{cases} \tag{20}$$

Where $1-\rho$ is the pheromone residual factor. $\tau_{ij}(t+1)$ and $\tau_{ij}(t)$ are the pheromone concentration from node i to node j at $(t+1)$-*th* iteration, *t-th* iteration, respectively. $\Delta\tau_{ij}(t)$ is the inverse of the cost of S_{ib}. The value of the pheromone trail is limiting between $[\tau_{\min}, \tau_{\max}]$ and the related formulas are defined in Equation(21) and (22).

$$\tau_{\max} = \frac{1}{\rho f(S_{sb})} \tag{21}$$

$$\tau_{\min} = \frac{(1 - \sqrt[n]{0.05})}{(\frac{n}{2} - 1)\sqrt[n]{0.05})} * \tau_{\max} \tag{22}$$

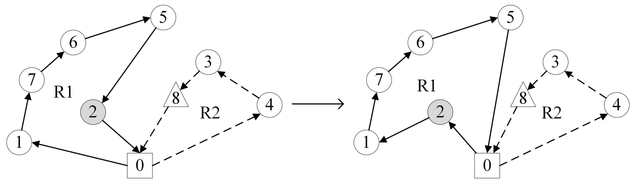
D. Neighborhood Structures

For the best ant obtained by each iteration, we employ three neighborhood operators in the VND procedure to improve the current ant. The neighborhood operators are described in the following.

(1) Relocate. A customer i is removed from a route and inserted into another position in the same or different route. In Fig.1(a), customer 2 of route $R1$ is moved and then placed in the position between depot and customer 1. As shown in Fig.1(b), customer 5 is removed from route $R1$ and inserted into the position after customer 3 in route $R2$. It is worth noting that only the customers can be used in the relocation operator.

(2) Two Points Swap. Two different customers swap their positions to change the positions of them. The operator occurs on the same route or the different routes. Fig.2 gives an example of the Two Points Swap operator. In Fig.2(a), customers 2 and customer 4 swap their positions on the same route $R2$. For two different routes $R2$ and $R1$, as shown in Fig.2(b), customer 2 on route $R1$ and another customer 5 on route $R2$ exchange their position. Similarly, this operator only operates on customer points.

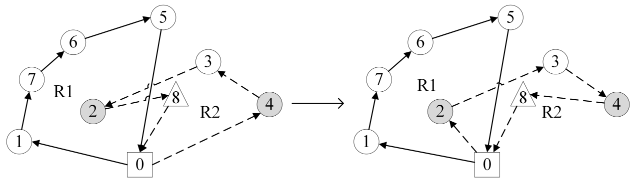
(3) 2-opt. When a 2-opt operator occurs in a route, two edges that are not adjacent are first chosen, and then the nodes between them are reversed. In Fig.3, two customers, 2 and 4 are selected, and the edge between 2 and 3 and the edge between 8 and 4 are broken. The nodes between the broken edges are reversed to 8, 5 and 3.



(a) Inter-route move

(b) Inter-route move

Fig. 1. example of Relocate operator



(a) Inter-route move

(b) Inter-route move

Fig. 2. example of Two Points Swap

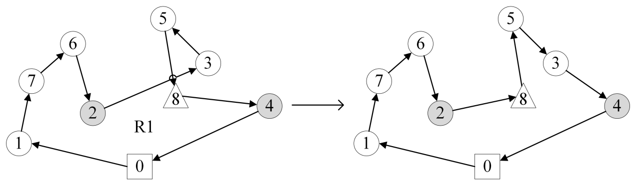


Fig. 3. example of 2-opt

E. Variable Neighborhood Descent Procedure

Variable Neighborhood Descent (VND) is a deterministic variant of Variable Neighborhood Search (VNS) developed by [18]. The basic idea of VND is to improve the current solution by constantly changing the neighborhood structure.

Algorithm 3 Variable Neighborhood Descent

Input: solution S, neighborhood structures NBS.
Output: the best solution S^*.
1: Initialize $S^* \leftarrow S$
2: $k \leftarrow \{1\}$
3: **repeat**
4: $\quad S' \leftarrow NBS_k(S^*)$
5: $\quad$ **if** $f(S') \leq f(S^*)$ **then**
6: $\quad\quad S^* \leftarrow S'$
7: $\quad\quad k \leftarrow 1$
8: $\quad$ **else**
9: $\quad\quad k \leftarrow k+1$
10: $\quad$ **end if**
11: **until** $k \leq |NBS|$
12: insert recharging stations into routes that violating battery constraint
13: **return** S^*

In HMMAS, VND is used to search for the local best solution after all ants have finished searching. The neighborhood operators are first executed without considering the battery amount constraint. After the neighborhood operator is executed, the solution will be checked. If the solution violates the constraints, the solution will be repaired to keep it feasible. The pseudo-code of the VND procedure is given in Algorithm 3.

In Algorithm 3, variable $|NBS|$ is the number of neighborhood operators. The main procedure of VND is step 3 $\sim$ step 11, which literately applies the neighborhood operator to find a new solution. When the new solution S' is better than the current solution, the current solution is updated and then returns the first neighborhood; otherwise, the search will enter the next neighborhood. When all the neighborhoods have been exploited, the VND search procedure terminates. It is worth noting that the neighborhood operator may change the order of the customers or the location of the charging station when it is executed, and the obtained neighborhood solution may violate the power constraint. As shown in step 12, if the new solution is not feasible, it should be repaired to be a feasible solution.

F. Charging Station Adjustment Strategy

When a solution is constructed or repaired, some charging stations are inserted into the routes to make the solution feasible. However, the position of the charging station may not be the best position. To decrease the total cost of routes, we design a charging station adjustment strategy to change the position of charging stations.

The main idea of this strategy is described in the following. First, the positions of all charging stations and depots at the route are firstly required in reversed order and put into a position list (PL). For example, route R is denoted as $\{0, a, s_1, b, c, s_2, d, 0\}$, where a, b, c are customers, s_1 and s_2 are stations, and PL is $\{7, 5, 2, 0\}$. The moving range of each station is obtained by PL. For example, if the station s_2 is shifted, its moving range in the route is between 7 and 2. Second, the moving cost for each charging station was calculated. The movement of the charging station is divided into two steps. The first step is to remove the station from the current route, and the second step is to insert new charging stations into the route without violating the battery constraint. So, the increasing cost of each move is the difference between CR_n and CR_o, in which CR_n is the cost of the new route after moving, and CR_o is the cost of the original route. Finally, the charging station that has the lowest increasing cost is selected to be moved. According to the above rule, one or multiple charging stations may be shifted to decrease the objective value of the solution.

IV. EXPERIMENTAL STUDY

To validate the advantages of our HMMSA method, we selected 14 instances from E-CVRP benchmark instances [21]. The scale of the problem is between 29 and 212. The instances consist of set E, set F, and set M, which are generated on the famous CVRP instances of Christofides and Eilon [23], Christofides et al. [24], and Fisher [25].

First, we describe the parameter settings of the algorithms. Then, we compare the results of HMMAS with existing algorithms ACS [12], MMAS [13], and VNS [21] on 14 E-CVRP instances. Finally, we analyze the efficiency of the algorithmic components of our hybrid heuristic, which is the VND procedure and charging station adjustment (CSA).

The proposed algorithm and comparison algorithms were coded in Python 3.9. All experiments were executed on a personal computer with the following parameters: Intel(R) Core (TM) i7-10700 CPU @ 2.90 GHz, 16.0 GB RAM. Meanwhile, all the algorithms were executed in 10 independent times for each instance.

A. Parameter setting

For HMMAS, ACS and HMMAS, the main parameters are the same. The number of iterations and population size are set to 300 and 25, respectively. The pheromone evaporation ρ, pheromone important factor α and heuristic information important factor β are 0.1, 1 and 2, respectively. The values of the max threshold and min threshold are set to 0.03 and 0.001, respectively. The parameter settings of the objective function are as follows: $c_1 = 100$, $c_2 = 1$, $c_3 = 1$.

B. Results and Comparison with existing algorithms

To evaluate the performance of our method, we compared the results of HMMAS with existing algorithms ACS, M-MAS and VNS on 14 E-CVRP instances. The results are shown in Table I and Table II. In the Table I, column *Best* indicates the *Best* cost value of each algorithm. Column *Gap* indicates the deviation calculated by the formula (23), where *BKS* is the best result of the above four methods. In Table II, column *Mean* and column *Time* represent the average solution and the average running time of 10 runs obtained by the algorithm, respectively. Column *Std* is the standard deviation of each algorithm.

$$Gap = \frac{(BKS - Best)}{BKS} * 100\% \tag{23}$$

As shown in in Table I, it can be seen that HMMAS is more competitive than the three existing algorithms, which has the lowest average total cost. The HMMAS method finds 13 optimal solutions in 14 benchmark instances and its success rate is 92.85%. Compared with ACS and MMAS,

TABLE I
COMPARISON RESULTS OF BEST SOLUTION AND GAP OBTAINED FOUR METHODS

Instance	BKS	ACS		MMAS		VNS		HMMAS	
		Best	Gap(%)	Best	Gap(%)	Best	Gap(%)	Best	Gap(%)
E-n29-k4-s7	1341.24	1367.51	1.96	1354.83	1.01	1374.96	2.51	1341.24	0.00
E-n30-k3-s7	1794.34	1887.03	5.17	1875.01	4.50	1879.36	4.74	1794.34	0.00
E-n35-k3-s5	1640.11	1801.87	9.86	1733.76	5.71	1722.39	5.02	1640.11	0.00
E-n37-k4-s4	2563.29	2690.27	4.95	2613.74	1.97	2572.80	0.37	2563.29	0.00
E-n60-k5-s9	1862.60	2271.57	21.96	1947.15	4.54	1936.76	3.98	1862.60	0.00
E-n89-k7-s13	2562.99	3042.79	18.72	2647.33	3.29	2617.19	2.11	2562.99	0.00
E-n112-k8-s11	3053.50	3821.00	25.14	3148.23	3.10	3099.61	1.51	3053.50	0.00
F-n49-k4-s4	2219.39	2301.19	3.69	2348.77	5.83	2219.39	0.00	2246.41	1.22
F-n80-k4-s8	1019.20	1185.88	16.35	1040.70	2.11	1036.95	1.74	1019.20	0.00
F-n140-k7-s5	3813.06	4677.25	22.66	3916.94	2.72	4141.94	8.63	3813.06	0.00
M-n110-k10-s9	3227.16	3850.04	19.30	3347.61	3.73	3666.35	13.61	3227.16	0.00
M-n126-k7-s5	3492.73	4317.58	23.62	3653.90	4.61	3980.62	13.97	3492.73	0.00
M-n163-k12-s12	4072.50	4991.05	22.55	4154.93	2.02	4181.00	2.66	4072.50	0.00
M-n212-k16-s12	5359.03	6394.77	19.33	5428.33	1.29	5430.01	1.32	5359.03	0.00
avg	2715.80	3185.70	15.38	2800.80	3.32	2847.10	4.44	2717.73	0.09

TABLE II
COMPARISON RESULT OF MEAN SOLUTION AND AVERAGE EXECUTION TIME

Instance	ACS			MMAS			VNS			HMMAS		
	Mean	Std	Time(s)	Mean	Std	Time(s)	Mean	Std	Time(s)	Mean	Std	Time(s)
E-n29-k4-s7	1415.69	26.85	19.39	1358.28	3.05	11.89	1419.49	23.97	2.54	1342.95	0.00	26.13
E-n30-k3-s7	1944.09	39.12	19.59	1895.32	18.34	11.76	2029.97	100.77	4.22	1810.41	9.66	24.79
E-n35-k3-s5	1842.04	16.20	22.77	1753.04	16.98	17.00	1755.92	38.82	5.53	1663.95	34.73	34.15
E-n37-k4-s4	2781.36	39.07	24.61	2638.48	16.97	16.93	2646.68	54.29	7.36	2605.22	16.74	77.03
E-n60-k5-s9	2333.28	42.09	44.42	1984.58	26.98	31.32	2119.36	98.24	29.22	1956.98	37.88	104.26
E-n89-k7-s13	3151.02	70.67	68.05	2718.76	45.60	59.62	2684.44	50.99	123.96	2622.98	41.65	263.19
E-n112-k8-s11	3872.04	44.61	90.88	3212.00	45.42	77.96	3149.35	35.88	184.64	3139.80	53.26	391.80
F-n49-k4-s4	2483.67	79.76	31.01	2434.55	55.02	20.94	2298.21	85.49	9.51	2189.36	23.22	50.09
F-n80-k4-s8	1206.12	15.59	55.18	1074.64	16.88	39.33	1189.23	83.24	54.46	1205.30	13.42	199.89
F-n140-k7-s5	4781.42	86.26	104.44	3998.96	54.22	102.81	4590.91	242.86	117.18	3903.00	58.97	646.84
M-n110-k10-s9	3951.97	60.00	85.49	3426.28	45.24	84.94	3759.98	73.75	120.26	3252.81	63.40	436.13
M-n126-k7-s5	4406.18	68.17	96.36	3729.44	42.84	84.84	4172.63	104.37	83.95	3665.98	85.79	652.16
M-n163-k12-s12	5150.62	89.29	144.70	4263.20	69.53	142.08	4263.05	59.66	326.77	4314.09	47.65	955.54
M-n212-k16-s12	6571.02	99.58	211.63	5644.02	89.04	215.45	5522.67	70.39	600.16	5685.84	50.89	2479.84
avg	3277.89	55.52	72.75	2866.54	39.01	65.49	2971.56	80.20	119.27	2811.33	38.38	452.99

HMMAS improve on average by 14.69% and 2.97%, respectively. When compared to the single-solution metaheuristic, which is VNS, the HMMAS algorithm decreases the total cost on average by 4.54%. From the results in Table II, the HMMAS also has the best average standard deviation value. The average standard deviation of HMMAS is 38.38, which is lower than the ACS algorithm 17.14, MMAS algorithm 0.63 and VNS algorithm 41.82. The average computation time of the proposed algorithm is a little higher than that of other comparison algorithms. The maximum-scale problem instances can also be solved effectively by the HMMAS algorithm within 45 minutes. On the whole, the HMMAS algorithm can find better solutions than other algorithms in a reasonable time.

C. Performance Analysis of Hybrid Components

As mentioned above, we introduced a variable neighborhood descent procedure (VND) and a charging station adjustment strategy (CSA) two-hybrid components into our proposed algorithm. In order to verify the effectiveness of the hybrid components of HMMAS, we designed two methods based on basic MMAS. One is the hybrid method of MMAS and CSA, namely MMASA. The other is to combine MMAS with VND, namely MMASB. We executed MMAS and two newly designed methods on seven benchmark instances from set E.

The experimental results of our proposed algorithm and three comparison methods are shown in Table III. The description of column *Best*, column *BKS* and column *Gap* are the same as above. As seen from Table III, we can observe that the proposed algorithm has the best performance and can obtain all the best solutions. When we employ the CSA in MMAS, the average total cost decreases from 2800.80 to 2796.63. It shows the designed CSA strategy can improve the quality of solutions. The VND procedure is also effective. It can decrease the average total cost 61.24 compared to

TABLE III
EXPERIMENTAL RESULTS OF DIFFERENT METHODS

Instance	MMAS			MMASA			MMASB			HMMAS		
	Best	Std	Gap(%)	Best	Std	Gap(%)	Best	Std	(%)	Best	Std	Gap(%)
E-n29-k4-s7	1354.83	3.05	1.01	1343.56	6.28	0.17	1341.24	4.33	0.00	1341.24	0.00	0.00
E-n30-k3-s7	1875.01	18.34	4.50	1844.51	31.23	2.80	1808.92	23.70	0.81	1794.34	9.66	0.00
E-n35-k3-s5	1733.76	16.98	5.71	1710.33	18.52	4.28	1642.73	34.81	0.16	1640.11	34.73	0.00
E-n37-k4-s4	2613.74	16.97	1.97	2621.99	10.90	2.29	2593.48	11.30	1.18	2563.29	16.74	0.00
E-n60-k5-s9	1947.15	26.98	4.54	1915.21	41.18	2.82	1909.79	24.48	2.53	1862.60	37.88	0.00
E-n89-k7-s13	2647.33	45.60	3.29	2665.55	21.15	4.00	2562.97	36.45	0.00	2562.99	41.65	0.00
E-n112-k8-s11	3148.23	45.42	3.10	3115.14	58.06	2.02	3074.55	38.44	0.69	3053.50	53.26	0.00
F-n49-k4-s4	2348.77	55.02	4.56	2326.92	38.46	3.58	2262.16	45.32	0.70	2246.41	23.22	0.00
F-n80-k4-s8	1040.70	16.88	2.11	1037.68	13.94	1.81	1019.63	10.50	0.04	1019.20	13.42	0.00
F-n140-k7-s5	3916.94	54.22	2.72	3921.39	70.15	2.84	3822.61	84.52	0.25	3813.06	58.97	0.00
M-n110-k10-s9	3347.61	45.24	3.73	3332.57	59.44	3.27	3264.74	61.64	1.16	3227.16	63.40	0.00
M-n126-k7-s5	3653.90	42.84	4.61	3667.25	42.92	5.00	3517.66	69.87	0.71	3492.73	85.79	0.00
M-n163-k12-s12	4154.93	69.53	2.02	4176.74	50.55	2.56	4086.19	69.03	0.34	4072.50	47.65	0.00
M-n212-k16-s12	5428.33	89.04	1.29	5474.03	84.58	2.15	5447.27	64.73	1.65	5359.03	50.89	0.00
avg	2800.80	39.01	3.23	2796.63	39.10	2.83	2739.57	41.37	0.73	2717.73	38.38	0.00

TABLE IV
COMPARISON RESULT OF DIFFERENT COMPONENTS USED IN MMAS

Methods	Algorithm Modules			Average	MTime(s)	MGap(%)	MCV(%)
	MMAS	VND	CSA				
MMAS	✓			2800.80	65.49	3.23	1.29
MMASA	✓		✓	2796.63	69.12	2.83	1.32
MMASB	✓	✓		2739.57	330.11	0.73	1.41
HMMAS	✓	✓	✓	2717.73	452.99	0.00	1.33

MMAS. When these two components are both used in MMAS, that is, our proposed algorithm, the improvement value of the average total cost is 83.08 compared to basic MMAS. The results show that the two components in our proposed algorithm are effective.

Further, we calculate the average execution time, average standard deviation and coefficient of variation of comparison methods. Table IV gives the results of different components composition. In table IV, columns *Average*, *MTime* and *MGap* represent the average of the best solution, execution time, and deviation of all instances, respectively. Column *MCV* is the average value of the coefficient of variation (CV), and the value of *CV* is calculated by the formula (24), where *Std* and *mean* represent standard deviation and mean value.

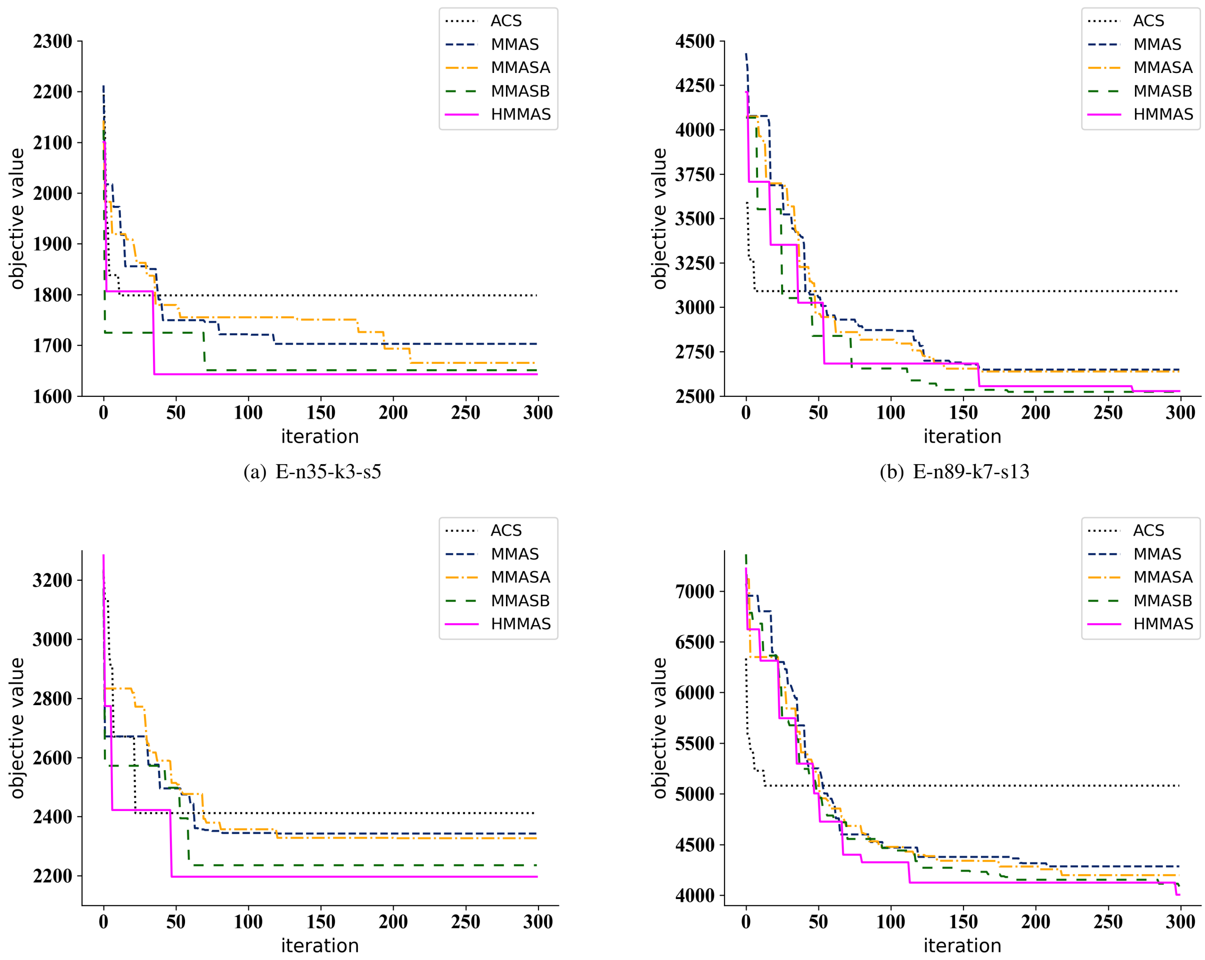
$$CV = \frac{Std}{Mean} * 100 \tag{24}$$

From Table IV, we can observe that CSA and VND components can improve the global optimization capability of MMAS. The data shows that the average total costs of MMASA, MMASB and HMMAS decrease by 0.15%, 2.19% and 2.97% compared to MMAS, respectively. So, the two components can improve the optimization ability of the algorithm. In addition, we also found that the CV values of all methods are between 1.2 and 1.5, indicating that the hybrid components have little effect on the stability. In view of average computation time, MMASA, MMASB and HMMAS increase a little. Even so, it is still acceptable because the algorithm can find a higher-quality solution.

D. Convergence Analysis of HMMAS

This section discusses the convergence of HMMAS by employing four E-CVRP instances having different scales. The instances are E-n35-k3-s5, E-n89-k7-s13, F-n49-k4-s4, and M-n163-k12-s12. Fig. 4 reports the convergence curves of the ACS, MMAS, MMASA, MMMSB, and HMMAS algorithms on these instances. The X-axis and Y-axis indicate the number of iterations and the optimization objective value, respectively.

In Fig. 4, we can see a steep declining trend in the convergence curve of HMMAS. In this case, the value of the objective function can rapidly drop and stabilize within a few iterations. It confirms that the proposed algorithm can quickly and effectively find the optimal solution, and exhibit faster convergence speed and accuracy. In Fig. 4 (a), the convergence curve of the MMASA algorithm remains constant for a specific period of iterations and subsequently continues to decrease. The charging station adjustment process causes the algorithm to jump out of the local optimum, resulting in a continuous decrease in the curve. The results demonstrate that MMASA has stronger global search capabilities and that the CSA strategy designed in this article is effective. In four instances, MMASB converges to the optimal value of about 60, 50, 150, and 180 generations, respectively. It converges faster than other algorithms and has better optimization capabilities, which proves that the VND module has a significant effect on improving the optimization capability of the algorithm.



(a) E-n35-k3-s5 (b) E-n89-k7-s13 (c) F-n49-k4-s4 (d) M-n165-k12-s12

Fig. 4. Convergence curves of four instances

It can be clearly seen from Fig. 4 that under the same number of iterations, the objective function value obtained by the proposed algorithm decreases faster and smaller, which shows that the convergence speed of HMMAS is faster. It has the advantage of finding better solutions. In conclusion, the HMMAS algorithm can effectively reach approximate optimal or global optimal, and its improvement strategies are also effective.

V. Conclusion

In this paper, we addressed the Electric Capacitated Vehicle Routing Problem with variable energy consumption rate (E-CVRP), where the variable energy consumption rate of vehicles exhibits a linear correlation with their load. Then, we proposed a hybrid max-min ant system algorithm, namely HMMAS, to solve it. After all ants complete a round of search, the variable neighborhood descent procedure is used as a local search process to optimize the iterative optimal solution. This procedure not only helps to enhance search capability but also accelerates convergence speed. In addition, the charging scheduling of routes is further improved by the charging station strategy. We conducted some experiments on 14 benchmark instances and the results demonstrate the effectiveness of our proposed algorithm.

In addition, we also test the performance of hybrid components in our proposed algorithm. The results show that introducing a VND procedure in the basic max-min ant system method can enhance the search ability of the algorithm. When we introduce the charging station adjustment strategy into the algorithm, the algorithm can obtain better solutions by shifting the position of charging stations. We also analyze the convergence curves of five methods on four instances, which demonstrates that the proposed HMMAS algorithm has fast convergence speed and good optimization ability.

In the future, we will do some work to improve the efficiency and generality of our proposed algorithm. Furthermore, with the application of electric vehicles in more and more fields, some new problems have emerged, such as time window, no-linear charging, and random demand, which have become new constraints of E-CVRP. This will be a promising research topic to solve these new variants of E-CVRP.

References

[1] X. Yin, W. Chen, J. Eom, L. E. Clarke, S. H. Kim, P. L. Patel, S. Yu, G. P. Kyle, "China's transportation energy consumption and CO_2 emissions from a global perspective," *Energy Policy*, vol. 82, pp. 233-248, 2015.

[2] R. G. Conrad and M. A. Figliozzi, "The Recharging Vehicle Routing Problem," in *Proceedings of the 2011 Industrial Engineering Research Conference*, 2011.

[3] S. Erdoğan and E. Miller-Hooks, "A Green Vehicle Routing Problem," *Transportation Research Part E: Logistics and Transportation Review*, vol. 48, no. 1, pp. 100-114, 2012.

[4] M. Schneider, A. Stenger and D. Goeke, "The Electric Vehicle-Routing Problem with Time Windows and Recharging Stations," *Transportation Science*, vol. 48, no. 4, pp. 500-520, 2014.

[5] D. Goeke and M. Schneider, "Routing a mixed fleet of electric and conventional vehicles," *European Journal of Operational Research*, vol. 245, no. 1, pp. 81-99, 2015.

[6] J. Lin, W. Zhou and O. Wolfson, "Electric vehicle routing problem," *Transportation Research Procedia*, vol.12, pp. 508-521, 2016.

[7] M. Keskin and B. Catay, "Partial recharge strategies for the electric vehicle routing problem with time windows," *Transportation Research Part C: Emerging Technologies*, vol. 65, pp. 111-127, 2016.

[8] A. Montoya, C. Guéret, J. E. Mendoza and J. G. Villegas, "The electric vehicle routing problem with nonlinear charging function," *Transportation Research Part B: Methodological*, vol. 103, pp. 87-110, 2017.

[9] M. Keskin, B. atay and G. Laporte, "A simulation-based heuristic for the electric vehicle routing problem with time windows and stochastic waiting times at recharging stations," *Computers & Operations Research*, vol.125, 105060, 2021.

[10] I. Kucukoglu, R. Dewil and D. Cattrysse, "The electric vehicle routing problem and its variations: A literature review," *Computers and Industrial Engineering*, vol.161, 107650, 2021.

[11] M. Thymianis, A. Tzanetos, E. Osaba, G. Dounias and J. D. Ser, "Electric Vehicle Routing Problem: Literature Review, Instances and Results with a Novel Ant Colony Optimization Method," in *2022 IEEE Congress on Evolutionary Computation (CEC)*, pp. 1-8, 2022.

[12] M. Dorigo and L. M. Gambardella, "Ant colony system: a cooperative learning approach to the traveling salesman problem," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 53-66, 1997.

[13] T. Stützle and H. H. Hoos, "MAXMIN Ant System," *Future Generation Computer Systems*, vol. 16, no. 8, pp. 889-914, 2000.

[14] C-Y. Lee, Z-J. Lee, S-W. Lin and K-C. Ying, "An enhanced ant colony optimization (EACO) applied tocapacitated vehicle routing problem," *Applied Intelligence*, vol. 32, no. 1, pp. 88-95, 2010.

[15] L. Wu, Z. He, Y. Chen, D. Wu and J. Cui, "Brainstorming-Based Ant Colony Optimization for Vehicle Routing With Soft Time Windows," *IEEE Access*, vol. 7, pp. 19643-19652, 2019.

[16] Z. Liu, X. Zuo, M. Zhou, W. Guan and Y. Al-Turki, "Electric Vehicle Routing Problem With Variable Vehicle Speed and Soft Time Windows for Perishable Product Delivery," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 6, pp. 6178-6190, 2023.

[17] M. Mavrovouniotis, G. Ellinas and M. Polycarpou, "Ant Colony optimization for the Electric Vehicle Routing Problem," in *2018 IEEE Symposium Series on Computational Intelligence (SSCI)*, pp. 1234-1241, 2018.

[18] Y. H. Jia, Y. Mei and M. Zhang, "A Bilevel Ant Colony Optimization Algorithm for Capacitated Electric Vehicle Routing Problem," *IEEE Transactions on Cybernetics*, vol. 52, no. 10, pp. 10855-10868, 2022.

[19] Li, H B, Zhang, X X, Fu, S and Hu, Y Y, "A Hybrid Algorithm Based on Ant Colony Optimization and Differential Evolution for Vehicle Routing Problem," *Engineering Letters*, vol. 29, no. 3, pp. 1201-1211, 2021.

[20] A. Froger, J. E. Mendoza, O. Jabali and G Laporte, "Improved formulations and algorithmic components for the electric vehicle routing problem with nonlinear charging functions," *Computers & Operations Research*, vol. 104, pp. 256-294, 2019.

[21] M. Mavrovouniotis, C. Menelaou, S. Timotheou, G. Ellinas, C. Panayiotou and M. Polycarpou, "A Benchmark Test Suite for the Electric Capacitated Vehicle Routing Problem," in *2020 IEEE Congress on Evolutionary Computation (CEC)*, pp. 1-8, 2020.

[22] N. Mladenović and P. Hansen, "Variable neighborhood search," *Computers & Operations Research*, vol. 24, no. 11, pp. 1097-1100, 1997.

[23] N. Christofides and S. Eilon,"An algorithm for the vehicle-dispatching problem," *Journal of the Operational Research Society*, vol. 20, no. 3, pp. 309-318, 1969.

[24] N. Christofides, A. Mingozzi, and P. Toth, "Exact algorithms for the vehicle routing problem, based on spanning tree and shortest path relaxations," *Mathematical Programming*, vol. 20, no. 1, pp. 255-282, 1981.

[25] M. Fisher, "Optimal solution of vehicle routing problems using minimum k-trees," *Operations Research*, vol. 42, no. 4, pp. 626-642, 1994.