Supplementary Material of "A Bi-level Ant Colony Optimization Algorithm for Capacitated Electric Vehicle Routing Problem"

Ya-Hui Jia, Yi Mei, *Senior Member, IEEE,* and Mengjie Zhang, *Fellow, IEEE*

Algorithm A $enumerate(\mathit{\Gamma}', sn, \mathbf{\Pi})$

Input: a route without charging station $\mathit{\Gamma}'$, the number of charging stations should be considered sn, candidate station list $\mathbf{\Pi} = \{\mathit{\Pi}_0, \ldots, \mathit{\Pi}_{|\mathit{\Gamma}'|-2}\}$

Output: an electricity-feasible route $\mathit{\Gamma}''$

```
1:  S_{0→sn−1} = 0, P_{0→sn−1} = 0;
2:  k = |Π|; Γ'' = ∅
3:  call the following recursive function;
4:  function recursion(mlen = k, nlen = sn)
5:      for i = (|Π| − mlen) → (|Π| − nlen) do
6:          for j = 0 → |Π_i| − 1 do
7:              S_{sn−nlen} = Π_{i,j};
8:              P_{sn−nlen} = i;
9:              if nlen > 1 then
10:                 recursion(|Π| − 1 − i, nlen − 1);
11:             else
12:                 insert the stations in S into Γ' according to the positions in P
13:                 if Γ' is better than Γ'' then
14:                     Γ'' = Γ';
15:                 end if
16:             end if
17:         end for
18:     end for
19: end function
20: return Γ'';
```

I. Enumeration Method

In this section, the $enumerate$ function that is used in Algorithm 3 of the paper is introduced in Algorithm A. First, we maintain a list S to store the stations to be inserted and a list P to store the positions where these stations should be inserted (line 1). The final solution is initialized empty (line 2). Then, we call a recursive function to try all the possible combinations which contains sn stations (line 3). All the arguments and variables in the function $enumerate$ are the global variables to the function $recursion$. The $recursion$ takes two arguments $mlen$ and $nlen$ that represents how many possible positions are left and how many stations are left to be inserted, respectively. These two arguments are initialized as $k = |\mathbf{\Pi}|$ and sn meaning that there are still k possible positions in $\mathit{\Gamma}'$ that have not been tried, and there are still sn stations left to be inserted (line 4). For each position $(\mathit{\Gamma}'_i, \mathit{\Gamma}'_{i+1})$, we try all the candidate stations in $\mathit{\Pi}_i$ (line 5-10). If the algorithm has generated a combination containing sn stations, we insert these stations in S into $\mathit{\Gamma}'$ to check whether it is better than $\mathit{\Gamma}''$ (line 11-12). If so, we update $\mathit{\Gamma}''$ (line 13-15). Finally, when all the combinations containing sn stations are already tried, $\mathit{\Gamma}''$ is returned.

II. Decomposition of CEVRP into CVRP and FRVCP

In this section, we first show the formulations of CEVRP and its sub-problems CVRP and FRVCP. Then, why BACO is a bi-level optimization algorithm instead of a two-stage optimization algorithm is explained.

A. Problem Formulation

In the main manuscript, the formulation of CEVRP has been give. However, due to the page limit, how this problem is treated as a bi-level optimization model and how it is decomposed into CVRP and FRVCP are not shown. Here, we will show the definitions of CVRP and FRVCP and analyse their relationships CEVRP.

Generally, a bi-level optimization problem can be defined as:

$$\min_{\mathbf{x}_u \in X_u, \mathbf{x}_l \in X_l} \quad F(\mathbf{x}_u, \mathbf{x}_l) \tag{1}$$

$$\text{s.t.} \quad G(\mathbf{x}_u, \mathbf{x}_l) \leq 0 \tag{2}$$

$$\min_{\mathbf{x}_l \in X_l} \quad f(\mathbf{x}_u, \mathbf{x}_l) \tag{3}$$

$$\text{s.t.} \quad g(\mathbf{x}_u, \mathbf{x}_l) \leq 0 \tag{4}$$

Corresponding to this definition, CEVRP is decomposed into two sub-problems. The upper level is CVRP and the lower level is FRVCP. Therefore, the decision variable of the upper-level sub-problem $\mathbf{x}_u$ represents the service orders of customers, and the decision variable of the lower-level sub-problem $\mathbf{x}_l$ represents the recharging schedules of vehicles.

1) Formulation of CEVRP: Before giving the formulation of CVRP and FRVCP, the formulation of CEVRP is shown here again to facilitate explanation. A CEVRP is defined on a fully connected weighted undirected graph $G = (V, E)$. $V = \{0\} \cup I \cup \hat{F}$ represents the set of nodes in the graph. 0 is the index of the depot. I represents the customers. Each customer i has a fixed cargo demand c_i. $\hat{F}$ is an extended set of charging stations that contains β_i copies of each charging station $i \in F$. $E = \{(i,j)|i,j \in V, i \neq j\}$ is the set of arcs. Each arc (i,j) is associated with a weight representing the distance between i and j, denoted as d_{ij}. Each EV has a maximum capacity of cargo demand C and a maximum battery capacity Q. The consumption rate of the battery is denoted as h. For each arc

(i,j), an EV will consume the amount $h \cdot d_{ij}$ of the battery to traverse it. u_i and y_i represent the remaining carrying capacity and remaining battery level of an EV when it arrives at node $i \in V$. The mathematical definition of CEVRP is given as follows:

$$\min\ f(\mathbf{x}) = \sum_{i \in V, j \in V, i \neq j} d_{ij} x_{ij}, \tag{5}$$

s.t.

$$\sum_{j \in V, i \neq j} x_{ij} = 1, \forall i \in I, \tag{6}$$

$$\sum_{j \in V, i \neq j} x_{ij} \leq 1, \forall i \in \hat{F}, \tag{7}$$

$$\sum_{j \in V, i \neq j} x_{ij} - \sum_{j \in V, i \neq j} x_{ji} = 0, \forall i \in V, \tag{8}$$

$$u_j \leq u_i - c_i x_{ij} + C(1 - x_{ij}), \forall i \in V, \forall j \in V, i \neq j, \tag{9}$$

$$0 \leq u_i \leq C, \forall i \in V, \tag{10}$$

$$y_j \leq y_i - h d_{ij} x_{ij} + Q(1 - x_{ij}), \forall i \in I, \forall j \in V, i \neq j, \tag{11}$$

$$y_j \leq Q - h d_{ij} x_{ij}, \forall i \in \hat{F} \cup \{0\}, \forall j \in V, i \neq j, \tag{12}$$

$$0 \leq y_i \leq Q, \forall i \in V, \tag{13}$$

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

where (6), (7), and (8) represent the constraints that each customer should be served exactly once and each copy of a station can be visited at most once. (9) and (10) represent the capacity constraint. (11), (12), and (13) represent the electricity constraint.

2) Formulation of CVRP: Defining the node set V' as the subset of V that does not contain charging stations $\hat{F}$, the formulation of CVRP is show as follows:

$$\min\ f(\mathbf{x}) = \sum_{i \in V', j \in V', i \neq j} d_{ij} x_{ij}, \tag{15}$$

s.t.

$$\sum_{i \in V'} x_{ij} = 1, \forall j \in I, \tag{16}$$

$$\sum_{j \in V'} x_{ij} = 1, \forall i \in I, \tag{17}$$

$$\sum_{i \in V'} x_{0i} = K, \tag{18}$$

$$\sum_{i \in V'} x_{i0} = K, \tag{19}$$

$$u_j \leq u_i - c_i x_{ij} + C(1 - x_{ij}), \forall i \in V', \forall j \in V', i \neq j, \tag{20}$$

$$0 \leq u_i \leq C, \forall i \in V', \tag{21}$$

$$x_{ij} \in \{0, 1\}, \forall i \in V', \forall j \in V', i \neq j, \tag{22}$$

where (16), (17), (18), and (19) are the constraints that each customer should be served exactly once and vehicles departure from the depot. K is the number of vehicles that can be either a fixed number or a variable. (18) and (19) are exactly the same as (9) and (10).

3) Formulation of FRVCP: Suppose a capacity-feasible route $\Gamma = [v_0, v_1, \ldots, v_{nr}, v_{nr+1}]$ where $v_0 = v_{nr+1} = 0$ represents the depot and $\{v_1, \ldots, v_{nr}\}$ represents nr customers in this route. Denoting the set $\{v_0, \ldots, v_{nr}\}$ as V'', the sub-problem FRVCP of CEVRP can be defined as:

$$\min\ \sum_{i=v_0}^{v_{nr}} ((\sum_{j \in \hat{F}} x_{ij}(d_{ij} + d_{j(i+1)})) + (1 - \sum_{j \in \hat{F}} x_{ij}) d_{i(i+1)}) \tag{23}$$

s.t.

$$\sum_{j \in \hat{F}} x_{ij} \leq 1, \forall i \in V'', \tag{24}$$

$$\sum_{i=v_0}^{v_{nr}} x_{ij} \leq 1, \forall j \in \hat{F}, \tag{25}$$

$$\sum_{j \in V'' \cup \hat{F}, i \neq j} x_{ij} - \sum_{j \in V'' \cup \hat{F}, i \neq j} x_{ji} = 0, \forall i \in V'' \cup \hat{F}, \tag{26}$$

$$y_j \leq yi - h d_{ij} x_{ij} + Q(1 - x_{ij}), \forall i \in V''/0, \forall j \in V'' \cup \hat{F}, \tag{27}$$

$$y_j \leq Q - h d_{ij} x_{ij}, \forall i \in \hat{F} \cup \{0\}, \forall j \in V'' \cup \hat{F}, \tag{28}$$

$$0 \leq y_i \leq Q, \forall i \in V'' \cup \hat{F}, \tag{29}$$

$$x_{ij} \in \{0, 1\}, \forall i \in V'' \cup \hat{F}, \forall j \in V'' \cup \hat{F}, \tag{30}$$

where (24), (25), and (26) show that each copy a station can be at most visited once. (27), (28), and (29) show the electricity constraints that are similar to (11), (12), and (13).

For a bi-level optimization algorithm, the overall objective is usually used to guide the upper-level decision making process. Even the upper-level sub-problem itself is a well-modeled problem, its objective is not considered like (1) and (3) showing. We only consider the overall objective (1) and the objective of the lower-level sub-problem (3) as a constraint. Correspondingly, although CEVRP is decomposed into CVRP and FRVCP in this paper, the objective of CVRP (15) is never used since we have a overall objective (5) of CEVRP. The purpose of giving the whole formulation of CVRP is to show the relationship between the decision variables of CVRP and CEVRP and the similarity between their capacity constraints. After decomposition, we can see that the variables related to the routes among customers and the depot are the variables of CVRP, and the variables related to the routes between stations and customers are the variables of FRVCP. Thus, the whole decision space is decomposed.

B. Difference between Bi-level and Two-stage

As we understand it, the difference between a bi-level optimization and a two-stage optimization is whether the overall objective is decomposed. For a bi-level optimization algorithm, it will not decompose the overall objective. The upper-level decision maker will adjust decisions according to the overall objective. For a two-stage optimization algorithm, it will decompose the overall objective into two sub-objectives. The upper-level decision maker will make decisions according to its sub-objective. The hypothesis of a two-stage optimization algorithm is that the optimal sub-solution of the upper-level/first-stage sub-problem can lead to the optimal integrated solution of the whole problem. However, this hypothesis is not always correct when solving a bi-level optimization problem. Taking CEVRP as an example, if we have two capacity-feasible solutions Γ_1 and Γ_2 of the corresponding CVRP, and Γ_1 is better than Γ_2 in terms of the objective (15), after fixing them to be electricity-feasible, Γ_1 may be worse than Γ_2 in terms of the overall objective (5).

In BACO, we proposed a OS-MMAS algorithm to generate the capacity-feasible solutions of the corresponding CVRP, but we did not directly use the objective (15) to update the pheromone matrix. Instead, after fixing the solutions to be electricity-feasible by solve the lower-level sub-problem FRVCP, the value of the overall objective (5) is used to update the pheromone matrix of OS-MMAS. This design perfectly conform to the definition of the bi-level optimization. Thus, BACO is truly a bi-level optimization algorithm rather than a two-stage optimization algorithm.

III. Parameter Tuning

In ACO algorithms, α and β are two important parameters to adjust the influence of the pheromone value and the heuristic information. The conventional setting, i.e. $\alpha = 1$ and $\beta = 2$, is adopted not only by MMAS but also other ACO algorithms such as ACS and AS. This setting is verified to be effective when these ACO algorithms are used to solve the traveling salesman problem. Since MMAS is used in BACO to generate the giant tour for splitting which is similar to generating a solution for a TSP, we inherited the conventional setting of MMAS, i.e. $\alpha = 1$ and $\beta = 2$, for the sake of simplicity.

Here, we investigate how these two parameters would affect the performance of BACO. Since what really matters is the ratio between these two parameters rather than their real values, we follow the traditional tuning methods in ACO algorithms that α is fixed to 1 and different β values are tried. In this experiment, besides $\beta = 2$, another three values {0,1,5} are tried on four instances {X143, X352, X573, X749}. The experimental results are shown in the Fig. A.

We can get the following observations from the experimental results:

- From the perspective of the final objective value, $\beta = 2$ is only worse than $\beta = 5$ on X573.On the other three instances, $\beta = 2$ is either equivalent to or better than the other values.
- The convergence speed of BACO increases with the growth of β. This phenomenon is more obvious on large-scale instances than on small-scale instances. On X143, $\beta = 1, 2, 5$ basically has the same convergence speed. On X352, the convergence speed of $\beta = 1$ starts to slow down. On X573 and X749, the convergence speeds of BACO under different β values have clear difference.
- However, the fast convergence speed of $\beta = 5$ is not always helpful. On X573, $\beta = 5$ can obtain both a fast convergence speed and a good final result, but, on X749, the fast convergence speed leads to premature convergence rather than a good final result.
- Setting $\beta = 0$ is the worst choice that BACO basically did not find any better solutions than the initial solution.

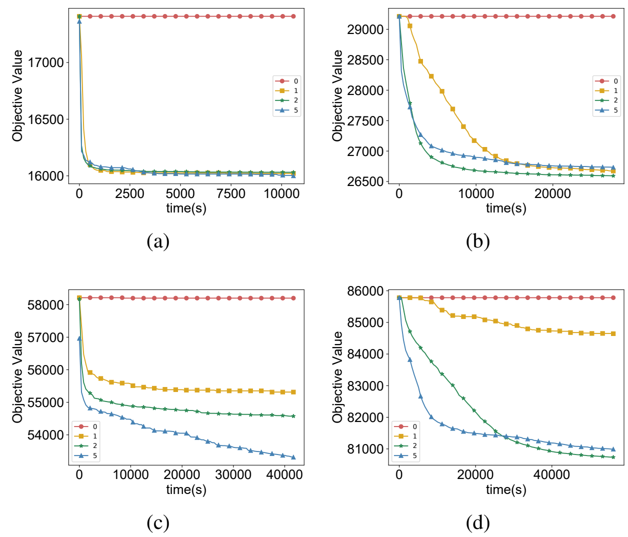

Fig. A. BACO performance under different β values. (a) X143, (b) X352, (c) X573, (d) X749.

Overall, the experimental results show a similar pattern to the experiments of using MMAS, AS, and ACS to solve TSP, which is in line with our expectation. Thus, setting $\beta = 2$ is appropriate for BACO.

IV. Full Results of Heuristic Comparison between FH, GH, and RH

The convergence curves of BACO using FH, GH, and RH on each instance are shown in Fig. B, corresponding to Fig. 7 of the manuscript.

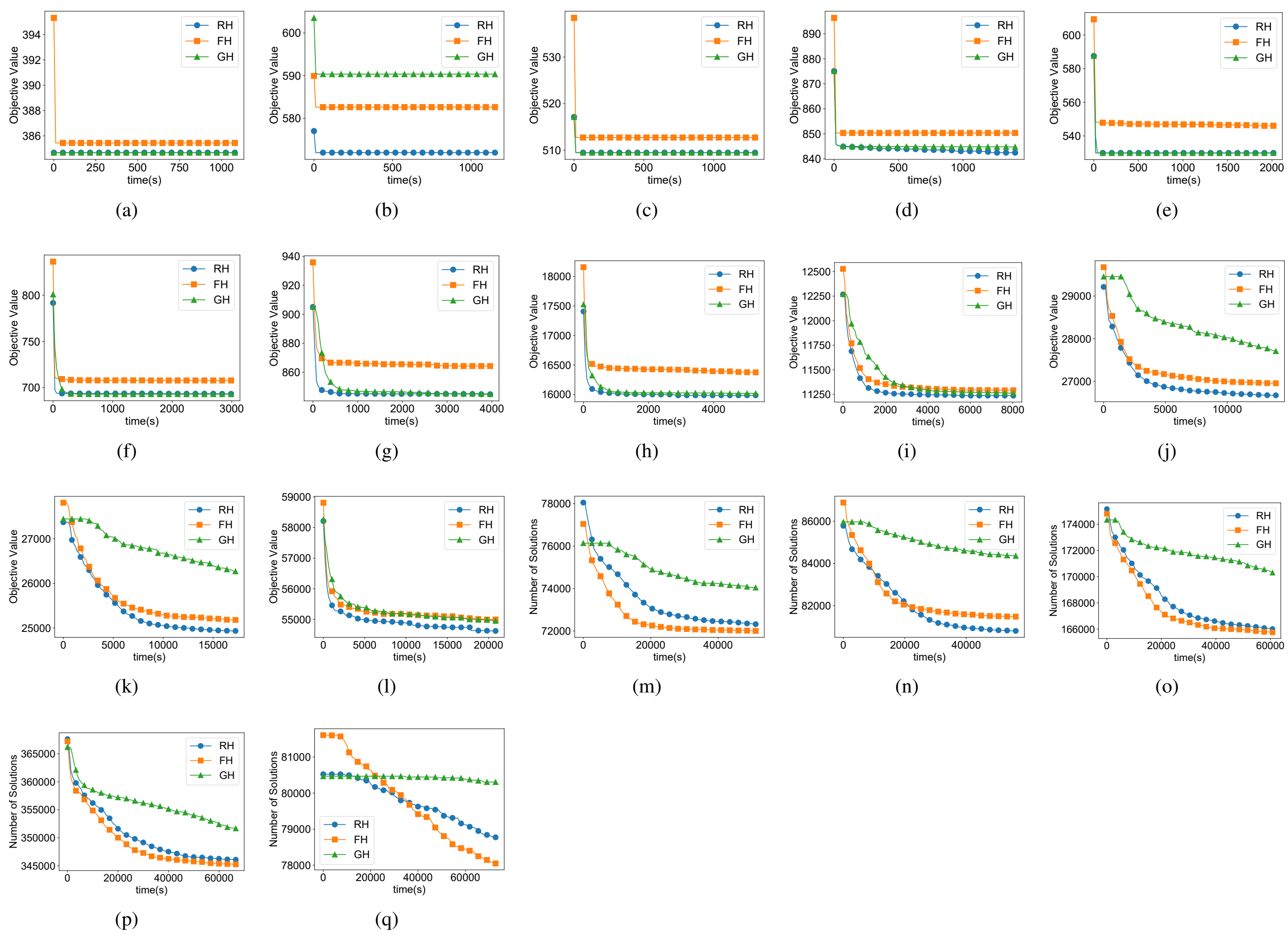

Fig. B. Convergence curves of using GH, FH, and RH in BACO.