A Two-phase Optimization Model for Autonomous Electric Customized Bus Service Design

Rongge Guo[1] and Wei Guan[2] and Saumya Bhatnagar[1] and Mauro Vallati[1]

Abstract— **Motivated by the requirements of highly effective customized bus (CB) service and by the rapid growth of autonomous electric vehicles (AEVs), this paper studies a new optimization model for the autonomous electric customized bus (AECB) service, aiming at minimizing operating costs and improving vehicles' efficient use. The proposed model contains two phases: (i) optimization of the vehicle routing, charging operation and passenger-to-vehicle assignment for the fixed travel demands, and (ii) re-optimization of the service according to real-time dynamic travel requests. A solution approach is developed to address the proposed model based on adaptive large neighborhood search (ALNS). The extensive empirical analysis, conducted by considering real-world data on a large-scale instance, demonstrates the efficiency of the proposed approach and the quality of the generated solutions.**

I. Introduction

The customized bus (CB) is an innovative and flexible public transit (PT) service, and it holds the promise to offer greater mobility and accessibility to groups of passengers with similar travel requests in both space and time [1]. With the advantages of better travel experience, congestion alleviation and emission saving, CB services have been recently introduced in a large number of cities in China.

Compared to the conventional bus, the CB system plans bus routes by aggregating travel demands collected from an online platform, in what can be seen as a demand-responsive transit (DRT) service [2]. However, travelers' spatial and temporal travel restrictions make it challenging to design the CB service. To address this challenge, simplifying assumptions have been considered, for instance by assuming fixed travel requests [3], thus the CB service design is treated as the static problem [4]. However, such simplifications lead to limited capabilities to cope with real-time requests. Alternative approaches to deal with real-time requests of CB system consider the perspective of adding new request to already planned routes [5]. However, inserting the new demands into the current routes leads to an increase of driving time, which may affect the crew schedule, and even dissatisfy the labor union regulation. In real-world practice, the operator assigns a new vehicle to serve the real-time requests, but this results in higher operating costs, since the crew cost dominates the vehicle costs. Such issues are preventing a fruitful and effective exploitation of the CB system, and there is the need for innovative approaches to unleash the full potential of the CB system.

The introduction of autonomous vehicles (AV) can provide the opportunity to overcome the issues of the CB system by eliminating the crew costs of drivers, and by removing the schedule constraints of drivers [6]. In fact, the use of AVs has been widely investigated in the context of car-sharing systems, as they support automated driving and allocation based on the real-time information, it means the vehicles can follow the fleet operation policies and maximize their fleet profitability [7]. Further, there is a growing interest in shared autonomous electric vehicles (SAEV), since the EV technology is the promising way to reduce the carbon footprint of transportation. Further, the integration of AVs and EVs makes it possible to automatize the battery management and charging process, which is helpful to eliminate range anxiety in human-driven EVs [8]. Besides, the smart optimization of SAEVs can improve the charging performance with less charging waiting events, resulting in an increased number of served trips [9].

Leveraging on the recent results in SAEVs, showing the improvement in vehicles' efficiency and served trips, and considering the performance benefits of AEVs in terms of labor cost and vehicle dispatching, in this paper we investigate the use of AEVs to address the CB service design problem, and we propose a solution to serve both the fixed and the real-time travel demands. More specifically, this work proposes a two-phase optimization model to minimize the operating cost of autonomous electric customized bus (AECB) system, where the operations of serving the fixed and real-time travel demands are defined as static and dynamic phases, respectively. In the static phase, the proposed model aims to simultaneously optimize the vehicle routes, passenger-to-vehicle assignment and charging operation based on the fixed spatial-temporal restrictions. In the dynamic phase, all the decisions are re-optimized taking advantage of the capabilities of AECBs, and guaranteeing compliance with the real-time travel demands. Herein, the dynamic phase maximizes the utilization of the AECB fleet, and improves the operation efficiency of the overall CB system. To deal with the proposed model, we introduce an approach based on the adaptive large neighborhood search (ALNS) algorithm, and we present dedicated destroy-repair operators that allow to efficiently search the high-quality solution space. The effectiveness of the proposed techniques is demonstrated using large-scale instances and historical real-world data. Compared against existing human-driven CB service, the proposed AECB system can lead to more than 20% reduction

[1]Rongge Guo and Saumya Bhatnagar and Mauro Vallati are with the University of Huddersfield, Huddersfield, United Kingdom. Rongge Guo and Mauro Vallati were supported by a UKRI Future Leaders Fellowship [grant number MR/T041196/1]. r.guo@hud.ac.uk, s.bhatnagar@hud.ac.uk, m.vallati@hud.ac.uk

[2]Wei Guan is with Beijing Jiaotong University, Beijing, China. weig@bjtu.edu.cn

in operating cost.

The remainder of this paper is organized as follows. Section II introduces the problem description, and the mathematical formulation is proposed in Section III. Section IV presents the solution algorithm. Section V reports the computational results of the experiment. Finally, Section VI gives the conclusions.

II. Problem Description

This section describes the addressed problem and the involved variables.

The AECB network is defined as a directed graph $G = (V, A)$, where V is the set of vertices so that $V = S \cup F \cup O$. Let S be the set of stations representing the origins and destinations (OD) of travel demands, F be the set of charging stations and O be the set of depots. A is the set of arcs, each of which has an associated distance d_{ij} and travel time t_{ij}.

The on-demand AECB problem is characterized to serve two types of passengers, namely fixed passenger groups $P = \{1, 2, \ldots, \mid P \mid\}$ submitted before the vehicle departure time t_0 and real-time passenger groups $P^{'} = \{\mid P+1 \mid, \mid P+2 \mid, \ldots, \mid P+P^{'} \mid\}$ that arrive occasionally over the service time, defined by a finite horizon $T = [t_0, \mid T \mid]$. Each passenger group represents a number of passengers $Num_{r,s}$ that have similar travel plans, including an origin r and a destination s, a corresponding preferred time window $[ear_r, lat_r]$, $[ear_s, lat_s]$, and service time ser_r, ser_s.

Fig. 1 presents the temporal nature of the AECB service process. For each passenger group, let t_p be the time that group p submits the request. At time t_0, the system has full information about the spatial-temporal requests of groups with submitted timestamp $t_p \leq t_0$ and no information about the future groups ($t_{p+1} > t_0$). After t_0, the time interval h is set to collect the occasional travel demands. Let t denote the end-time of each time interval, then the AECBs are re-planed to serve new passengers at t, until t is greater than T. To summarize, the AECB service design problem addressed here can be described as a two-phase procedure: (i) in the static phase ($t \leq t_0$), The central system optimizes the AECB service according to the fixed requests; (ii) in the dynamic phase ($t > t_0$), the system re-plan the service for occasional requests at t. At timestamp $t \geq t_0$, a fleet of homogeneous AECBs with same capacity is assigned to serve passengers. To guarantee the modest profit goal, the served passengers of each AECB need to be greater than the minimum load factor $Load$, while the number of visited stations for one trip should be less than the maximum visiting station N_{max} to reduce the passenger in-vehicle time.

Let E be the full battery capacity of AECBs, the discharge of the battery occurs when the AECB moves across arcs. Theoretically, it is reasonable to assume that AECBs travel with constant speed, and therefore consume linearly the amount $u \cdot d_{ij}$ of the remaining battery capacity that is greater than the cut-off remaining battery value E^{*}. The recharging time incurred by an AECB depends on its energy level e_i^k when arriving at the charging station. We assume that the battery of AECBs is charged with a constant current-constant voltage (CC-CV) scheme. Specifically, the state of charge (SOC) increases linearly with constant charging rate g in CC phase; then it increases concavely with charging time in CV phase after reaching the threshold $\hat{SOC}$ of CC phase [10]. Let convert SOC into the battery capacity: let $\hat{E}$ be the maximum achievable battery capacity in CC phase, $f(t)$ represents the function of remaining battery capacity with respect to charging time in the CV phase. The time required to charge an AECB from e_i^k to E can be calculated by:

$$t_e^k = \begin{cases} (\hat{E} - e_i^k)/g + f^{-1}(E) & e_i^k \leq \hat{E} \\ f^{-1}(E) - f^{-1}(e_i^k) & e_i^k > \hat{E} \end{cases} \quad (1)$$

One key idea behind our study is to identify the available AECBs for serving in both the static and dynamic phases. Each AECB $k \in K$ has a physical location $pos^k(t)$ and a state $\beta_k(t)$ at timestamp t. AECB k can be in one of three states $\beta_k(t) \in 1, 2, 3$ that are idle $K_I(t) = \{K \mid \beta_k(t) = 1\}$, en-route $K_E(t) = \{K \mid \beta_k(t) = 2\}$ and charging $K_C(t) = \{K \mid \beta_k(t) = 3\}$, respectively. The idle state corresponds to AECBs with full charge in depot. The en-route state corresponds to AECBs moving to pick up and drop off passengers. The charging state corresponds to AECBs at charging station. In the static phase, all AECBs with idle state are available for serving and can be considered. In the dynamic phase, besides the idle AECBs, the nearest charging AECBs are considered available to respond to real-time requests. Without the labor cost and crew schedule, the re-planning of charging AECBs is one of the major elements that can lead to cost-saving solutions.

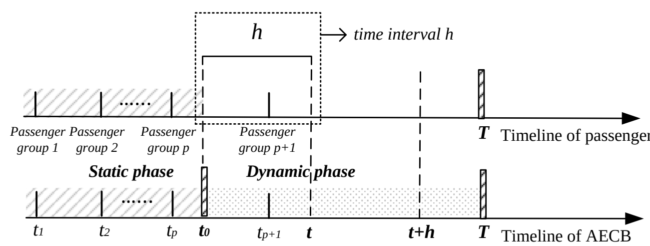

Fig. 1. Depiction of temporal aspects of AECB service design.

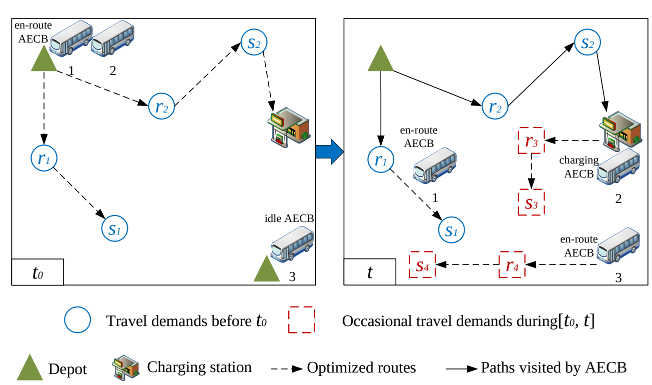

Fig. 2. AECB service design for timestamps t_0 (left) and t (right).

Fig. 2 gives an example of the service over the two phases. At $t_0 (t_0 = t - h)$, two nearest AECBs (1 and 2) are planned to serve passengers groups 1 and 2 (defined by the corresponding origins r and destinations s), the AECB state turns to en-route $\beta_k(t) = 2$. At time t, the nearest AECB 2 with charging state and AECB 3 are assigned to serve the new requests (3 and 4) that arrived during h, the state of AECB 3 changes to en-route $\beta_k(t) = 2$.

It is worth reminding that both the static and dynamic phases involve significant optimization-related decisions, including AECB route plan, charging operation and passenger-to-vehicle assignment, defined by variables $x_{i,j}^k$ (equal to 1 if arc (i, j) on the route of AECB k), e_i^k (remaining battery capacity of an AECB at vertex i) and $y_{r,s}^k$ (equal to 1 if passenger group $r \to s$ is served by AECB k), respectively. To determine the position of vehicles, an intermediate variable ξ_i^k (AECB k departure time at vertex i) is defined.

III. A TWO-PHASE OPTIMIZATION MODEL FOR AECB

This section is devoted to present the two phases of the optimization model for the AECB service.

A. AECB static phase

The static phase focuses on the travel requests submitted before the service start time, and can be seen as an AECB routing problem with fix travel demands. The mathematical model of static phase is formulated as a mixed-integer program as follows. The objective is to minimize the operating cost, that is related to the departure cost, travel distance and charging time as defined in (2), where c_f,c_d and c_e are fixed departure cost per vehicle, travel cost per km and charging cost per min.

$$\begin{aligned} min\, z = c_f \sum_{i \in O} \sum_{j \in S} \sum_{k \in K_I} x_{i,j}^k + c_d \sum_{(i,j) \in A} \sum_{k \in K_I} x_{i,j}^k d_{i,j} \\ + c_e \sum_{i \in S} \sum_{j \in F} \sum_{k \in K_I} x_{i,j}^k t_e^k \end{aligned} \tag{2}$$

s.t.

$$\sum_{i \in V} \sum_{k \in K_I} x_{i,j}^k = 1,\, j \in S \tag{3}$$

$$\sum_{i \in O} \sum_{j \in S} x_{i,j}^k = \sum_{i \in O} \sum_{j \in S \cup F} x_{j,i}^k = 0,\, k \in K_I \tag{4}$$

$$\sum_{i \in S} x_{i,j}^k \geq 1,\, j \in F, k \in K_I \tag{5}$$

$$\sum_{i \in V} x_{i,j}^k = \sum_{i \in V} x_{j,i}^k = 0,\, j \in S, k \in K_I \tag{6}$$

$$\sum_{k \in K_I} y_{r,s}^k = 1,\, r, s \in S \tag{7}$$

$$y_{r,s}^k \leq \sum_{j \in S} x_{r,j}^k,\, r \in S, k \in K_I \tag{8}$$

$$y_{r,s}^k \leq \sum_{i \in S} x_{i,s}^k,\, s \in S, k \in K_I \tag{9}$$

$$Load \leq \sum_{r \in S} \sum_{s \in S} y_{r,s}^k Num_{r,s} \leq cap,\, k \in K_I \tag{10}$$

$$\begin{aligned} \xi_j^k + B \geq \xi_i^k + (t_{i,j} + ser_j) x_{i,j}^k + B \cdot x_{i,j}^k, \\ i \in S \cup O, j \in V, k \in K_I \end{aligned} \tag{11}$$

$$\begin{aligned} \xi_j^k + B \geq \xi_i^k + (t_{i,j} + t_e^k) x_{i,j}^k + B \cdot x_{i,j}^k, \\ i \in F, j \in V, k \in K_I \end{aligned} \tag{12}$$

$$ear_i \leq \xi_i^k \leq lat_i,\, i \in S, k \in K_I \tag{13}$$

$$t_e^k = \begin{cases} (\hat{E} - e_i^k)/g + f^{-1}(E) & e_i^k \leq \hat{E} \\ f^{-1}(E) - f^{-1}(e_i^k) & e_i^k > \hat{E} \end{cases} \tag{14}$$

$$\begin{aligned} E^* \leq e_j^k \leq e_i^k - (u \cdot d_{i,j}) x_{i,j}^k + B(1 - x_{i,j}^k), \\ i, j \in S, k \in K_I \end{aligned} \tag{15}$$

$$E^* \leq e_j^k \leq E - (u \cdot d_{i,j}) x_{i,j}^k, i \in F \cup O, j \in S, k \in K_I \tag{16}$$

$$\sum_{i \in V} \sum_{j \in V} x_{i,j}^k \leq N_{max},\, k \in K_I \tag{17}$$

Constraint (3) enforces the connectivity of stations visits, and constraints (4)-(5) handle the connectivity of visits to depots and recharging stations. Constraint (6) ensures the flow conservation. Constraint (7) indicates that all passengers are assigned to exactly one vehicle. Constraints (8)-(9) represent the fact that an AECB can transport travel demand only if it goes through both such demand's origin and destination. Constraint (10) ensures that in-vehicle passengers do not exceed the vehicle capacity and at the same time meet the minimum load requirement $Load$. Constraints (11)-(12) guarantee time feasibility for arcs leaving stations, depot and recharging visits, B is a large number. Constraint (13) enforces that every station is visited within its time window. Constraints (14) encode the charging time function. Constraints (15) and (16) ensure that the battery charge never falls below a minimum threshold. Finally, constraint (17) forces the number of stations visited by an AECB to be below a given maximum N_{max}.

B. AECB dynamic phase

In the dynamic phase, new travel requests from occasioanl travelers are accepted during the time interval h. The AECB network is re-optimized to ensure the maximized utilization of available AECBs, which means the charging AECBs can be re-scheduled after charging operation for the real-time requests.

The dynamic routing phase addresses the occasional requests by serving them with charging AECBs ($k \in K_C$) or by launching an idle AECB ($k \in K_I$) for the new passengers. To determine the state of traveling AECB $\beta_k(t)$ at re-optimizing time stamp t, it is necessary to determine the positions of AECBs with regards to the arrival and departure times $\xi_i^k(t-h)$ at every vertex obtained in the static phase. Let $\mu_i^k(t-h)$ denote the arrival time at timestamp $t-h$ that can be calculated with the following equation.

$$\mu_i^k(t-h) = \begin{cases} \xi_i^k(t-h) - ser_i & i \in S \\ \xi_i^k(t-h) - t_e^k(t-h) & i \in F \\ \xi_i^k(t-h) & i \in O \end{cases} \tag{18}$$

After acquiring the arrival times, the positions and states of traveling AECBs can be obtained as follows:

1) if $x_{i,j}^k = 1$ and $t > \mu_j^k(t-h)\,(i \in S \cup F, j \in O)$, then $pos^k(t) = v_j\,(j \in O)$, the state changes from en-route to idle $\beta_k(t) = 1\,(k \in K_I)$, the AECB is available for serving new requests with the associated departure cost;
2) if $x_{i,j}^k = 1$ and $t \in [\xi_i^k(t-h), \mu_j^k(t-h)]\,(i, j \in V)$, then the AECB is on the arc (i,j), $pos^k(t) = null$, the state remains en-route $\beta_k(t) = 2\,(k \in K_E)$, and the AECB is unavailable;
3) if $x_{i,j}^k = 1$ and $t \in [\mu_j^k(t-h), \xi_j^k(t-h)]\,(i \in V, j \in S)$, then $pos^k(t) = v_j\,(j \in S)$, the vehicle state remains en-route $\beta_k(t) = 2(k \in K_E)$, and the AECB is unavailable;
4) if $x_{i,j}^k = 1$ and $t \in [\mu_j^k(t-h), \xi_j^k(t-h)]\,(i \in V, j \in F)$, then $pos^k(t) = v_j\,(j \in F)$, the state changes from en-route to charging $\beta_k(t) = 3\,(k \in K_C)$, the AECB is available after charging.

According to the vehicle state, it is possible to re-optimize the AECB operation at time stamp t with the model proposed in the static phase by considering also the charging AECBs, that is, modifying $k \in K_I$ into $k \in K_I \cup K_C$ for constraints (2)-(17).

In addition, constraint (19) is added into the model to enforce that the departure time of the AECB with charging state equals to the departure time obtained at $t-h$, which ensures that the AECB leaves the charging station with full battery capacity E.

$$\xi_i^k(t-h) = \begin{cases} t & i \in O \\ \xi_i^k(t-h) & i \in F \end{cases} \tag{19}$$

IV. Solution approach

This section introduces a solution approach based on adaptive large neighborhood search (ALNS) algorithm to solve the AECB routing and rerouting problem that is NP-hard. ALNS has been successfully applied to the various VRP [11]. As the extension of LNS, ALNS iteratively improves an initial solution by destroying and repairing it. The possibilities of destroy and repair operators being chosen depend on their performance in the past iterations [12].

The proposed approach applies the ALNS algorithm for both static and dynamic phases to improve the quality of the generated solution. It means that ALNS is used at each timestamp t of the dynamic phase. To acquire the best solution at each timestamp, we first propose the initial solution generation method based on greedy algorithm; then, we design a number of paired destroy-repair operators to perform local search; Afterwards, the adaptive strategy is introduced to update the probability of each operator with the performance, while the simulated annealing algorithm is utilized to accept the solution ($x \to x'$, if $f(x') < f(x)$ with a probability $e^{f(x)-f(x'))/T}$) [11]. It is worth noting that the position pos^k and states β_k need to be identified at each timestamp, to acquire the available AECBs for serving. The solution algorithm is shown in Algorithm 1.

Algorithm 1 Solution Approach

Initialization: $\hat{t} \leftarrow t_0$
while $\hat{t} < T$ **do**
  $x \leftarrow InitialSolution$
  $x^* \leftarrow x$
  while the termination criterion is not satisfied **do**
    select destroy and repair operators with probabilities p^- and p^+
    $x_t \leftarrow Repair(Destroy(s))$
    if $accept(x_t, x)$ **then**
      $x \leftarrow x_t$
    end if
    if $f(x) > f(x^*)$ **then**
      $x^* \leftarrow x$
    end if
    update p^- and p^+
  end while
  $\hat{t} \leftarrow \hat{t} + h$
end while
Return: x^*

A. Initial solution generation

An initial solution is constructed by a greedy algorithm that produces a locally optimal solution in three steps: first, randomly select N_{max} vertices (paired origins and destinations) with floyd algorithm to generate a route; then confirm that no violation (time window and vehicle capacity constraints) occurs for new route, if any violation occurs, back to first step; afterwards check the remain battery at each vertex and insert the charging station into high-consumption routes. The construction of initial solution is presented in Algorithm 2.

B. Destroy and repair operators

We design and implement three destroy operators:

a) Random removal: This operator randomly selects n passenger groups to be removed from the current solution.

b) Worst removal: This operator first computes the cost savings produced by the removal of each group; n groups with maximum savings are then removed.

c) Charging removal: This operator first computes the electricity amount E_i produced by charging stations$(i \in F)$; if E_i is not greater than minimum electricity amount E_{min} , then the charging station i is removed.

Based on the defined destroy operators, we propose three corresponding repair operators to re-organize the solution:

d) Random insertion: At each iteration, each removal group is randomly inserted into the destroyed routes, until none can be inserted.

e) Greedy insertion: At each iteration, the best insertion cost is computed for each destroyed route and the removal group with the lowest insertion cost is inserted at its best position. The operator stops when none can be inserted.

f) Charging insertion: For the destroyed routes, finding the farthest vertex $i(i \in S)$ that can reach without the charging; the best insertion cost is then computed for the nearest charging station, the charging station with the lowest insertion cost is inserted after i. The operator stops when the solution satisfies all the constraints.

Algorithm 2 Construction of Initial Solution

Input: Passenger groups P with $[ear_r, lat_r]$ and $[ear_s, lat_s]$, $InitialSolution = \oslash$

```
while passenger groups are not served do
    choose two depots a and b, an idle AECB k ∈ K_I
    P' ← RandomSelect(P)
    L ← GenerateRoute(a, P', b)
    if (ear_r ≤ ξ_r^k ≤ lat_r)&&(ear_s ≤ ξ_s^k ≤ lat_s) then
        if Load ≤ Σ_{p∈P'} Num_p ≤ cap then
            if e_i^k ≤ E* then
                insert the nearest i(i ∈ F) into L
            end if
        end if
    end if
    InitialSolution ← InitialSolution + L
end while
```

Return: $InitialSolution$

V. Experiment and computational results

To assess the performance of the proposed approach, we conduct a series of large-scale experiments using conventional bus smart-card data as the travel demands. All the experiments are run on an Intel Core i5-5257 machine, with 2.70 GHz, and 8 GB RAM. All the compared algorithms are coded and executed in ILOG CPLEX and MATLAB R2016b.

A. Experiment design

The smart-card data (SCD) of conventional buses collected from 24th to 28th April 2017 in Beijing are acquired to estimate the spatial-temporal travel demands. The SCD of each passenger contains the ID, boarding and alighting station, and boarding and alighting time [13]. Preferred time windows of every vertex are defined by two timestamps as follows: each origin or destination corresponds to one timestamp in SCD, we assume that the first passenger's boarding time is $ear_i(i \in S)$ and last one's boarding time is $lat_i(i \in S)$ for this passenger group. According to the passenger travel patterns during the morning period, 36 vertices with 26 trip groups (300 passengers) from 7:00-9:00am are extracted as the fixed travel demands and occasional travel requests of experiment. In the dynamic phase, the time interval is set to 45 minutes.

In the implementation, we assume AECBs with 40-seat capacity, 20-people load requirement and 100kWh full battery capacity. Other parameters are given below: c_f =500¥/veh, c_d=15¥/km, c_e=10¥/min, N_{max}=6. The discharging rate is 2kWh/km, and the cut-off remaining battery is 20kWh. The recharging rate is 4kWh/min in CC phase, where the maximum achievable battery capacity is 80kWh [14]. The charging profile is approximated by a piece-wise liner function in CV phase [10]:

$$f(t) = \begin{cases} 80 + 2.5(t-20), & t \in [20, 22) \\ 85 + 1.875(t-22), & t \in [22, 30] \end{cases} \tag{20}$$

B. Solution approach comparison

This section presents several experimental results demonstrating the effectiveness of the proposed solution approach. The performance of ALNS is evaluated by comparing its output with the optimal solutions, which is found by the commercial solver CPLEX 12. Besides, ALNS is also compared with the tabu search (TS) algorithm that has proven to be effective in solving related problems [11]. It has been tested that ALNS can obtain the best price within 50 iterations, thus the iteration of ALNS and TS is 50 to ensure the validity of the comparison results.

Table I shows an overview of the solutions and the relative differences with regards to the results of CPLEX. For comparison sake, we report the best values obtained for 10 runs by the ALNS and TS. ALNS produces high-quality solution, with the lowest gap to CPLEX, yielding better solutions than TS. It can be explained that the destroy and repair operators generated achieve larger neighborhood than TS. Further, ALNS is capable of producing near-optimal solutions by requiring significantly less CPU-time: in our experiments, ALNS can generate solutions an order of magnitude faster than CPLEX. This also suggests that ALNS would be in the best position to provide high-quality solutions for large-scale instances, where CPLEX is likely to fail or to require too much time to be used in a real-world deployment.

TABLE I
Comparison of the performance achieved by CPLEX, TS, and ALNS.

	Cost (¥)		Distance (km)		Travel Time (min)		Charging time (min)		CPU time (min)	
		Diff.		Diff.		Diff		Diff.		Diff.
Cplex	10145.7		325.2		557.5		216.8		446	
TS	11444.6	9.6%	344.8	6.0%	603.0	4.8%	227.2	8.2%	49	-89.0%
ALNS	10744.9	2.9%	334.9	3.0%	574.1	2.5%	222.2	3.0%	32	-92.8%

C. Comparison to human-driven customized service

This section studies how AECBs perform with regards to human-driven electric customized bus (HECB). To compare two services for different types of passenger distributions, we generate three classes of instances, categorized similarly to the R, C, and RC Solomon instances [15]. R, C and RC correspond to random customer distribution, clustered customer distribution, and a mixture of both, respectively. Each problem instance has 26 trip groups with 300 passengers.

For HECB service, we assume that the CB operation in static phase is the same as AECB, while only the idle vehicles can be assigned to serve new passengers during dynamic phase due to the crew schedule. It means that the HECBs with idle state are available after identifying the vehicle positions, and the charging HECBs are not available for serving until they go back to depot. Thus, we use the model for static phase to optimize the service of each timestamp in dynamic phase.

Results are shown in Fig. 3. Overall, we observe that the savings in operating cost, travel distance and charging time consuming enabled by the effective exploitation of AECBs are very significant for each considered instance. The average savings are about 20.6%, 4.5% and 4.3% in operating cost, distance, and charging time, respectively. Notably, AECB can drastically reduce departure costs and the number of vehicles, when compared to HECB. This is particularly relevant to the crew schedule of HECB: AECBs are allowed to serve after charging operation, whereas HECBs are not. Flexible vehicle scheduling therefore decreases the costs for AECB service.

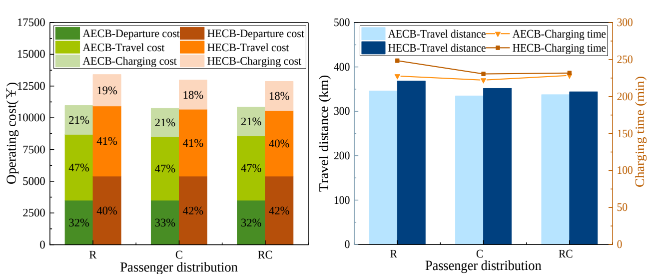

Fig. 3. Comparison results of AECB and HECB in terms of costs (left) and traveled distance (right).

VI. Conclusion

This paper introduces and analyzes the AECB service design problem in the presence of both fixed and real-time travel demands. A two-phase optimization model is proposed to determine the vehicle routing, passenger-to-vehicle assignment and charging operation in both static and dynamic phases. The aim is to optimize and re-optimize customized service with the benefits of AEVs in terms of non-labor cost, complete vehicle dispatching control and emission saving, to minimize operating costs.

The algorithmic approach presented in this paper leverages on the ALNS algorithm, exploiting effective destroy and repair operators. We perform an extensive experimental analysis by considering large-scale data, and demonstrate the capabilities of the presented algorithms with regards to optimal solvers and baselines. We also compare the AECB service with the HECB service: our analysis shows that the AECB can lead to significant savings in operating cost, travel distance and charging time consuming for different passenger distributions.

For future work, we plan to take into account the en-route AECBs for serving the real-time travel demands in the dynamic phase, and we are also interested in hybrid CB service design with AVs and HVs.

References

[1] X. Chen, Y. Wang, and X. Ma, "Integrated optimization for commuting customized bus stop planning, routing design, and timetable development with passenger spatial-temporal accessibility," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 4, pp. 2060–2075, 2021.

[2] R. Guo, W. Zhang, W. Guan, and B. Ran, "Time-dependent urban customized bus routing with path flexibility," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 4, pp. 2381–2390, 2020.

[3] M. Gong, Y. Hu, Z. Chen, and X. Li, "Transfer-based customized modular bus system design with passenger-route assignment optimization," *Transportation Research Part E: Logistics and Transportation Review*, vol. 153, p. 102422, 2021.

[4] W. Shu and Y. Li, "A novel demand-responsive customized bus based on improved ant colony optimization and clustering algorithms," *IEEE Transactions on Intelligent Transportation Systems*, 2022.

[5] C. Wang, C. Ma, and X. D. Xu, "Multi-objective optimization of real-time customized bus routes based on two-stage method," *Physica A: Statistical Mechanics and its Applications*, vol. 537, p. 122774, 2020.

[6] B. Loeb, K. M. Kockelman, and J. Liu, "Shared autonomous electric vehicle (saev) operations across the austin, texas network with charging infrastructure decisions," *Transportation Research Part C: Emerging Technologies*, vol. 89, pp. 222–233, 2018.

[7] M. W. Levin, K. M. Kockelman, S. D. Boyles, and T. Li, "A general framework for modeling shared autonomous vehicles with dynamic network-loading and dynamic ride-sharing application," *Computers, Environment and Urban Systems*, vol. 64, pp. 373–383, 2017.

[8] T. D. Chen, K. M. Kockelman, and J. P. Hanna, "Operations of a shared, autonomous, electric vehicle fleet: Implications of vehicle & charging infrastructure decisions," *Transportation Research Part A: Policy and Practice*, vol. 94, pp. 243–254, 2016.

[9] H. Miao, H. Jia, J. Li, and T. Z. Qiu, "Autonomous connected electric vehicle (acev)-based car-sharing system modeling and optimal planning: A unified two-stage multi-objective optimization methodology," *Energy*, vol. 169, pp. 797–818, 2019.

[10] M. Xu and Q. Meng, "Fleet sizing for one-way electric carsharing services considering dynamic vehicle relocation and nonlinear charging profile," *Transportation Research Part B: Methodological*, vol. 128, pp. 23–49, 2019.

[11] I. Žulj, S. Kramer, and M. Schneider, "A hybrid of adaptive large neighborhood search and tabu search for the order-batching problem," *European Journal of Operational Research*, vol. 264, no. 2, pp. 653–664, 2018.

[12] S. Ropke and D. Pisinger, "An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows," *Transportation science*, vol. 40, no. 4, pp. 455–472, 2006.

[13] R. Guo, W. Guan, A. Huang, and W. Zhang, "Exploring potential travel demand of customized bus using smartcard data," in *Proc. of IEEE ITSC*, 2019, pp. 2645–2650.

[14] L. Zhang, S. Wang, and X. Qu, "Optimal electric bus fleet scheduling considering battery degradation and non-linear charging profile," *Transportation Research Part E: Logistics and Transportation Review*, vol. 154, p. 102445, 2021.

[15] D. Taş, N. Dellaert, T. van Woensel, and T. De Kok, "The time-dependent vehicle routing problem with soft time windows and stochastic travel times," *Transportation Research Part C: Emerging Technologies*, vol. 48, pp. 66–83, 2014.