EVTour: Online Scheduling System for Tours with Multiple Destinations by One-Way EV Sharing

Naoki Shibata*, Tomoyuki Ueda*, Weihua Sun†, Minoru Ito*
*Nara Institute of Science and Technology
8916-5 Takayama, Ikoma City, Nara, Japan.
Email: {n-sibata, tomoyuki-u, ito}@is.naist.jp
†Shiga University, Banba, Hikone 522–8522, Japan
Email: akira-kawai@biwako.shiga-u.ac.jp

***Abstract*—In this paper, we propose an online scheduling system for tours by one-way electric vehicle (EV) sharing. The system takes reservations of tours with their multiple destinations, time windows and stay time for visiting each destination. It then tells the user the sequence and scheduling of visits to each destination, and which specific EVs to take. The system tells the user to change to another EV when the battery runs out. This enables users to move long distances without concerns about remaining battery. The system also calculates a schedule for moving the EVs to rebalance their distribution. In order to verify the effectiveness of our system, we evaluated it with simulations with 500 to 700 users. Our results showed that the proposed system improved the acceptance rate of reservations while reducing the relocation count(the number of EVs to be moved rebalance the distribution) by up to 15%.**

I. Introduction

Electric vehicles(EVs) are attracting attention given their environmental advantages over conventional gasoline- or diesel-powered vehicles. EVs have the advantages of less noise and higher power efficiency, and they can also be used as tools for demand response to level out spikes in electricity consumption. However, the typical cruising distance of a small EV is approximately 200km, and a quick recharge takes 30 minutes. The cruising distance is also affected by cruising speed, traffic congestion, road gradient, and use of an air conditioner. And to relieve a user from concerns about low battery, infrastructures would have to be built that allow recharging on the go.

On the other hand, carsharing allowing people car rentals for short periods of time is increasingly popular. Carsharing is already widespread in North America and Europe, where companies like Autolib'[1] and car2go[2] provide services for one-way carsharing. Autolib' is an electric car sharing service that was started in Paris. As of July 2014, over 2500 vehicles had been registered for the service, which has over 155000 registered subscribers. Autolib' offers over 4000 charging points in Paris. Users pick up and return a car at any station. Each car has on-board GPS capabilities and is tracked by the operations center.

In one-way carsharing, the distribution of the cars can get unbalanced, thus requiring personnel to move the cars between parking stations to rebalance the distribution. There has been much research on one-way carsharing. However, most of the existing researcher only considers use of conventional cars.

In this paper, we propose a practical online scheduling system for one-way EV sharing, i.e. our system manages schedules for many users and EVs, accepts incoming reservation requests and executes the scheduled events in parallel. A user makes a reservation request that includes multiple destination stations to visit and the time window and stay time for each station. The system calculates an order and scheduling of visits to each station that satisfy all specified conditions. The system indicates which EV the user should take to move between stations. The user is asked to use a different EV, or recharge, when running out of battery. This enables users to move long distances without concerns about remaining battery. The proposed system enables quickly calculating efficient schedules for a large number of users and EVs, while calculating the schedule for personnel who move the EVs to rebalance the distribution.

In order to verify the effectiveness of our system, we developed a simulator and performed evaluation with 500 to 700 users and 340 vehicles. We used a road map of Tokyo with real traffic patterns obtained from Japan Road Traffic Information Center. Our results showed that the proposed system can find a user schedule from each reservation request in less than 5 seconds on average. We also found that the system improved the acceptance rate of reservations by 4% compared to the existing method and reduced the number of EV relocations by 15%. The acceptance rate of reservation requests declined by only 3% at times of traffic congestion, and there was almost no user waiting time for a vehicle to arrive at each station.

II. Related Work

One-way carsharing is convenient for users, but it requires personnel to move cars between parking stations to balance the distribution of vehicles. There has been much research on carsharing. Some works address the reduction of environmental load through carsharing, while others focus on efficient rebalancing of vehicle distribution. In some research on rebalancing distribution, only service personnel move the cars; in other works, users are also utilized in some way to move the cars.

Barth et al. developed a simulation model of a shared vehicle system for evaluating operational issues such as vehicle availability, vehicle distribution, and energy management[3]. The authors applied this model to a resort community in Southern California, and their cost analysis showed that shared vehicle systems can be very competitive with present transportation systems. The same authors proposed two user-based relocation mechanisms called trip joining (or ridesharing) and trip splitting[4]. When the system realizes that the distribution of EVs is becoming imbalanced, these mechanisms urge users who have more than one passenger to take separate vehicles when more vehicles are needed at the destination station.

Febbraro et al. used discrete event systems to represent the complex dynamics of the carsharing system, and proposed a user-based methodology on the basis of an optimal relocation policy in a rolling horizon framework to maximize operator benefits by reducing the number of required staff to relocate vehicles among the stations and determines the minimum number of vehicles needed to satisfy system demand[5].

Kek et al. developed simulation models with operator-based relocation techniques[6]. They proposed performance indicators to evaluate the effectiveness of the different relocation techniques. The model has been validated by using commercially operational data from a local shared-use vehicle company. The simulation results showed that if the relocation technique is used, the system can afford a 10% reduction in car park lots and 25% reduction in staff strength, generating cost savings of approximately 12.8% without lowering the level of service for users.

Smith et al. addressed the problem of relocation by employing rebalancing drivers to drive vehicles from popular destinations to unpopular destinations[7]. They studied the problem of the rebalancing drivers themselves become unbalanced, and proposed a strategy for optimally routing the rebalancing vehicles and drivers so that stability is ensured while minimizing the number of rebalancing vehicles traveling in the network and the number of rebalancing drivers needed.

Bruglieri et al. addressed the relocation problem with one-way electric vehicle sharing[8]. They presented a new approach for the Electric Vehicle Relocation Problem, where cars are moved by personnel of the service operator to keep the system balanced.

Almost all existing works on the relocation problem do not consider use of electric vehicles, and some are not evaluated with realistic data on users and on the supply and demand of vehicles, nor on user movement or unexpected congestion. In this paper, we not only consider the relocation problem with one-way electric vehicle sharing, but also propose a method for calculating the order of stations to visit, and allow users to change the vehicles to ride at each station.

III. Overview of the System

In this paper, we propose a system for scheduling tours with one-way EV sharing. The system manages the schedules for users, EVs, and charging booths. To make a reservation, a new user is prompted to input multiple stations to visit, the final destination, the time window and the stay time at each station. If the system finds a schedule with which the user can visit all of the specified stations satisfying all the specified conditions, then the system accepts the request and incorporates the new schedule into the existing schedules for users, EVs, and charging booths. The system then gives the user the instructions for the tour. The proposed system tries to maximize the acceptance rate of reservation requests.

A. *Environment*

We assume that typical stations include large facilities like stores, railway stations, airports, and parking lots. Each station has a limited number of charging booths for EVs. Personnel at the station drive the EVs to other stations when instructed by the server. We also assume that the number of persons in a user group (e.g. a family) is limited and that a whole group of users can ride in a single EV. In this paper we do not consider cases in which a group of users divide up and ride multiple EV, or in which multiple groups ride in a single EV. Hereafter, we call a group of users simply a user.

A user tries to make a reservation via a website or a smartphone app. If the new reservation request can be accommodated, the server presents the schedule to the user. On the day of the tour, the user receives an IC card with which the user can start up the assigned EV. The server instructs each EV to allow only the user who has the specified IC card to start the assigned EV. Then, the user follows the schedule to visit all of the requested destinations. Each EV has a GPS receiver and a wireless communication device, and the server always knows the location of and the other information about each EV.

Instead of making a reservation in advance, a user can visit stations on the day of tour, and then make a reservation on the spot. Users can use their mobile phones or PCs or the fixed terminals installed at stations to make a reservation and receive instructions from the server.

B. *Usage model*

We now explain the entire process from making a reservation to the end of a tour.

Users making a reservation in advance: Up until the day prior to the tour, users can use their mobile phones to make a reservation via a web site or a smartphone app. The user is prompted to input the stations to visit, the final destination, the time window and the stay time at each station, alongside the payment information. The user is immediately informed of whether the request can be accepted. The user can try different routes with assistance by the computer, until the request can be accommodated. When the reservation request is accepted, the user also receives the arrival and departure times at each station. On the day of the tour, the server indicates upon departure a specific EV to use. Then, the user takes the common actions(describe below).

Users making a reservation on the spot: Users can visit stations on the day of the tour and use fixed terminals or mobile phones to make reservation. The user is immediately informed of whether the request can be accommodated. If the request is accepted, the user receives the arrival and departure

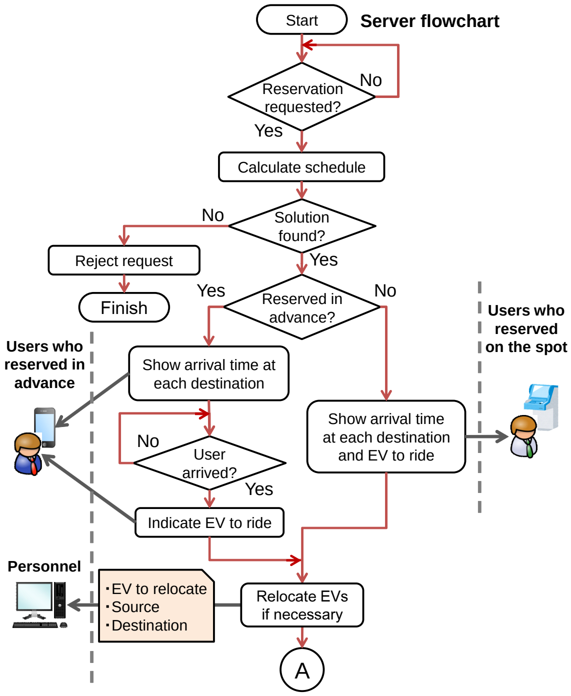


Fig. 1. Server flowchart for accepting reservation from a user

times at each station and the ID of the EV to take to the next destination. Then, the user takes the actions described in the next section of this paper.

The common actions: The user drives the specified EV to the next destination along the instructed route. When the user arrives at the next station, he/she is asked by the server to recharge the EV if necessary. In this case, the user moves the car to the specified charging booth. Then, he/she does what he/she planned to do in the vicinity of the station. At the scheduled departure time, the user checks via a mobile phone or a terminal the ID of the EV to be used. At this time, the user may be instructed to ride a different EV from the previous EV. If there is no EV available for the next trip due to delay of arrival or some other reason, the server asks the user to wait until a new EV becomes available. Upon arriving at the final destination, the user returns the IC card, and the tour finishes.

The personnel: They receive instructions from the server via the terminal at the station. An instruction contains the ID of the EV and the ID of the destination station. After the service person receives an instruction, he/she drives the specified EV to the destination. Upon arrival, he/she may be instructed to recharge the EV at a specific charging booth at the station.

C. Functions of the server

A server retains all user information, locations of the EVs, remaining battery and the status of each charging booth. It also manages schedules for all users, EVs and charging booths, and updates the schedules according to various events. The server has two major tasks. The first task is to find a user schedule that satisfies all conditions specified by a user. The other task is to find and maintain a relocation schedule to

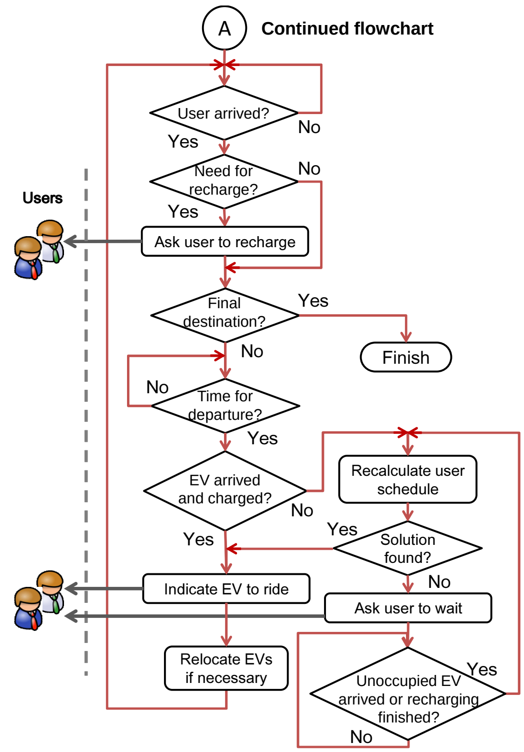


Fig. 2. Server flowchart for guiding the user through the tour

rebalance the distribution of EVs. In the following, we explain how the server finds the schedules and instructs the users and personnel.

Accepting requests from users: When the server receives a reservation request from a user, the server checks the schedules and tries to find a schedule that satisfies all conditions(Fig 1). It presents the schedule to the user if the request can be accommodated. The new user schedule is incorporated into the schedule plan of other users. The corresponding schedules for EVs and charging booths are updated at this time. At departure time, the server instructs the user of the ID of the EV to use for moving to the next station. The user then departs from the station, proceeding in accordance with the instructions from the server.

Guiding users during tours: After the user departs from a station, the server instructs the personnel to relocate cars. When the user arrives at the next station, the server checks the remaining battery of the EV, and asks the user to recharge, as stipulated in the schedule(Fig. 2). At the time of departure for the next station, the server indicates the ID of the EV for moving to the next station. If the scheduled EV is not available due to unexpected traffic congestion, the server starts rescheduling procedures.

Rescheduling procedure: If the scheduled EV for a user is not ready at departure time, the server tries to recalculate the schedule for the user. If a new solution is found, the server updates the schedules for the user, and informs the user of the substitute EV. If no solution is found, the server asks the user to wait until a new EV is ready. Since the server needs to update the schedules occasionally, at the time of reservation it indicates only the arrival and departure time for each destination; the specific IDs of EVs to take and other detailed information are indicated only upon departure.

Relocation procedure: The server periodically checks the scheduled number of spare EVs at each period of time. Spare EVs can be used to accept new incoming reservation requests, or substitute EVs that become unavailable due to delay of arrival. If the distribution of spare EVs is not balanced, the server updates the relocation schedule. When an EV departs from a station, the relocation is executed according to the schedule.

IV. Formulation of The Scheduling Problem

A. Input

The input consists of database input and user input.

Database input

- **Digital map:** A graph $G = (N, A)$ that represents the road network is given. Each node $n \in N$ and each link $a \in A$ in G represents an intersection, and a road segment, respectively. Each link has a weight called the *congestion coefficient*, and this is used for calculating the influence on the cruising distance and speed that are affected by traffic congestion, road gradient, and use of the air conditioner. The following functions are given, where $n_i, n_j \in N$ are neighboring nodes in G.
  - $dist(n_i, n_j)$ is the distance between n_i and n_j.
  - $speed(n_i, n_j)$ is the typical cruising speed between n_i and n_j.
  - $jam(n_i, n_j, t)$ is the coefficient for extra energy consumption associated with congestion between n_i and n_j at time t.
- **Stations:** The set of all stations $D \subseteq N$ is given. The following information on each station $d \in D$ is given.
  - The set of charging booths at the station.
  - The number of EVs initially located at the station.
- **Charging booths:** The set of all charging booths is given. The following information on each booth is given.
  - The station where the booth is installed.
  - The amount of energy that can be charged in a unit of time.
- **EVs:** The set E of all EVs is given. The following information is given corresponding to each EV $e \in E$.
  - The station where the EV is initially located.
  - The initial remaining battery.
  - The battery capacity.
  - The energy consumption per unit cruising distance.

User input: The set of all users is denoted by U. The following information is given corresponding to each user $u \in U$.

- Starting time of the tour
- First station
- Final station
- Set of all stations to visit
  - Time window of arrival at each station
  - Stay time at each station

B. Output

The server retains schedules for all users, all EVs and all charging booths. The server operates by managing and modifying these schedules according to the incoming reservation requests and arrival and departure of users at each station. Effectively, the output shown below constitutes a part of the input for the scheduling problems.

User schedule: A user schedule represents a tour schedule for the corresponding user, and this is a list of user schedule elements. Each element contains the following information.

- Station ID
- ID of EV for moving to the next station
- Arrival time
- Departure time

EV schedule: An EV schedule represents a schedule for moving users that day by the corresponding EV. It is a list of EV schedule elements. Each element contains the following information.

- User ID
- Station ID
- Charging booth ID
- Arrival time
- Departure time
- Remaining battery upon arrival
- Remaining battery upon departure

Charging booth schedule: A charging booth schedule is a schedule for the charging of EVs by the corresponding booth that day. The following is a list of charging booth schedule elements. Each element contains the following information.

- ID of the EV to charge
- Starting time of charging
- Finishing time of charging

C. Constraints

All schedules have to satisfy all of the following constraints.

- The remaining battery of each EV is positive upon arrival at each station.
- Each EV has to satisfy the condition of the time window given by the user for arrival at each station, considering the congestion coefficient.
- One EV has always to be assigned to each user.
- When an EV is to be charged, a charging booth has to be assigned to that EV.

We now explain the first constraint. Suppose that EV $e \in E$ moves from station $d_i \in D$ to $d_j \in D$ at time t. Let $n_k \in N$ denote the k-th node passed through, and let list $[n_0, n_1, ..., n_k]$ denote the list of all nodes that are passed through during the trip. Here, $n_0 = d_i$ and $n_k = d_j$. The consumed energy $C(d_i, d_j, e, t)$ can be calculated as follows.

$$C(d_i, d_j, e, t) = \frac{\sum_{q=0}^{k-1} dist(n_q, n_{q+1}) \cdot jam(n_q, n_{q+1}, t)}{\text{Energy consumption rate by EV } e} \quad (1)$$

Let $P(d_i)$ denote the remaining battery upon departure at d_i, and the following inequation has to be satisfied for each EV.

$$P(d_i) - C(d_i, d_j, e, t) > 0 \quad (2)$$

D. Objective function

The objective of the proposed system is to maximize the acceptance rate of reservation requests. The acceptance rate is the ratio of the accepted reservation requests to all reservation requests.

V. Scheduling Methods

We propose an algorithm for finding user schedules and an algorithm for setting relocation schedules as a part of the proposed method. The former algorithm tries to find a user schedule when a new reservation request is received. It checks all retained schedules and tries to find a new user schedule that can be accommodated. The algorithm for setting relocation schedules is executed periodically. It checks the EV schedules and the number of spare EVs at each station, and sets schedules for relocating EVs by modifying the EV schedules.

A. Algorithm for finding user schedules

This algorithm finds an order of stations to visit by which the user can visit all the requested stations within the specified time window. The algorithm must immediately find a schedule after the system receives a reservation request. This problem is the traveling salesman problem with additional conditions regarding energy consumption, charging time and time constraints at each destination. Thus, finding the exact solution may take exponential time. We therefore use a genetic algorithm(GA) to find an approximate solution within a realistic time frame.

Encoding of a candidate solution: A candidate solution is a list of the station IDs.

Fitness function: In the algorithm, the fitness function of a user schedule $s = [d_1, ..., d_k]$ for user $u \in U$ is defined by (3). This function is a weighted sum of the number of stations that can be visited satisfying all the constraints, and the total driving distance. $sat(u, d)$ is 1 if all the constraints described in IV-C are satisfied for user u at station d, and 0 otherwise. a and b are positive constants.

$$f(s) = a \sum_{i=1}^{k} sat(u, d_i) - b \sum_{i=1}^{k-1} dist(d_i, d_{i+1}) \quad (3)$$

Algorithm: The steps of the proposed genetic algorithm are as follows:

1) **Initialization:** Generate candidate solutions by shuffling the order of stations.
2) **Crossover:** One-point crossover is used to make a new candidate solution. Duplicated stations are removed.
3) **Mutation:** Stations are swapped probabilistically.
4) **Evaluation:** Calculate the fitness of each candidate solution according to Eq. (3).
5) **Selection:** Roulette wheel selection and the elite preservation strategy is used to select candidate solutions for the next generation.
6) Step 2 to 5 are repeated.

B. Algorithm for setting relocation schedules

This algorithm is executed periodically to prevent decline of the acceptance rate caused by unbalanced distribution of EVs. It creates new schedules for relocating EVs by calculating the time, source and destination stations for moving EVs. Hereafter, a margin at a station refers to the number of cars available for new reservation requests at the station. The system checks margins at stations for each hour and sets new schedules for relocation if necessary. The EV schedule is updated according to the new schedules. For each hour, the algorithm chooses the source and destination stations according to margins at stations, and then it creates schedules by executing the following two stages in the scheduling.

The first stage tries to shorten the distance of relocation. It finds source and destination stations that are next to each other, and schedules relocation between these stations. In detail, for each destination station d_i, the first stage algorithm checks if there is a source station next to d_i, and if there is such a station, the algorithm assigns the station with highest margin to d_i, and updates margins for those two stations.

The second stage utilizes simulated annealing to globally rebalance distribution of EVs. It tries to lower the variance of margins at all stations while keeping the distance of relocation short. In other words, it tries to distribute an excessive number of vehicles concentrated in small areas. A candidate solution is a list of pairs of IDs of source and destination stations. The destination ID can be NULL, which indicates no relocation schedule will be set for the corresponding source station. The fitness function $f(R, t)$ for a candidate solution R is shown in Eq.(4), where t is the time to relocate EVs. $var(t, D)$ and $dist(s_i, d_i)$ denote, respectively, the variance of margins at all stations after relocation and the distance of relocation for the i-th member of the list. a and b are positive constants.

$$f(R, t) = a \cdot var(t, D) + b \sum_{i=1}^{|R|} dist(s_i, d_i) \quad (4)$$

VI. Evaluation

In this section, we show the results of our evaluation with 500 to 700 users from the perspectives of execution time, acceptance rate with and without congestion, and relocation count.

A. Configuration

We used a map and congestion information of Tokyo, which is shown in Fig.3, provided by the Japan Road Traffic Information Center(JARTIC)[9]. This map covers approximately a 10km×10km area at the center of Tokyo. We converted

TABLE I
PARAMETERS FOR THE EXPERIMENTS

Parameter	Value
Number of nodes	68
Number of links	221
Number of stations	68
Number of charging booths per station	1
Number of initial EVs at each station	5
EV speed without congestion	40km/h
EV speed in light congestion	20km/h
EV speed in heavy congestion	10km/h
Charging time	120 min.
Energy consumption rate	5km/kWh
Battery capacity	16kWh
Number of users	500 to 700
Number of destinations per user	0 to 5
Stay time at a station	60 to 90 min.
Ratio of reservation in advance	10%

this map into a graph with 68 nodes and 221 links. The traffic information is provided on a JARTIC web page as a picture, which is updated hourly. The traffic information at each segment of the road map is indicated as a color of each link on the picture. We saved this picture every hour and manually converted the information (color of each link) to congestion attributes on the graph. Stations are placed at all intersections on the map, and we placed 5 EVs at each station when we started the simulation. The total number of EVs was 340. We generated the users who would make reservations in advance(10% of total users). We then started the simulation. The remaining users made reservations on the spot; these users were generated randomly during the time period for the simulation, which was from 8:00 to 18:00. The speed of the EVs changed according to the traffic congestion information. The other parameters for our simulation are shown in Table I. We developed our simulation program with Java, and ran our program on a PC with Windows 7 Professional x64 OS, Intel Core i7 920(2.66GHz) and 12GB memory. We compared the proposed method with the following methods:

- **No Relocation(NR):** Our proposed method minus relocation.
- **Maximum Allowance(MA):** The method proposed in [6]. This method relocates EVs from the station with largest margin to the station with lowest margin.

B. Average time to find a user schedule

Table II shows the average execution time for the algorithm to check if a new reservation request can be accommodated and then to generate a user schedule for this reservation. The results show that even when the number of users is 700, it takes less than 5 seconds to find a schedule.

TABLE II
AVERAGE TIME TO FIND A USER SCHEDULE

Number of users	500	600	700
Avg. time to find a user schedule	2716ms	3044ms	4106ms

Fig. 3. Map of Tokyo with congestion information

TABLE III
EXAMPLE OF RESERVATION

Item of reservation	Value
User ID	12
Departure time	08:06
ID of the first station	27
ID of the final destination	43
ID and stay time(min) for stations to visit	[(57, 72), (5, 78), (39, 75)]
Time window for arrival	10:06-11:06 (Station 39)

C. Example schedules

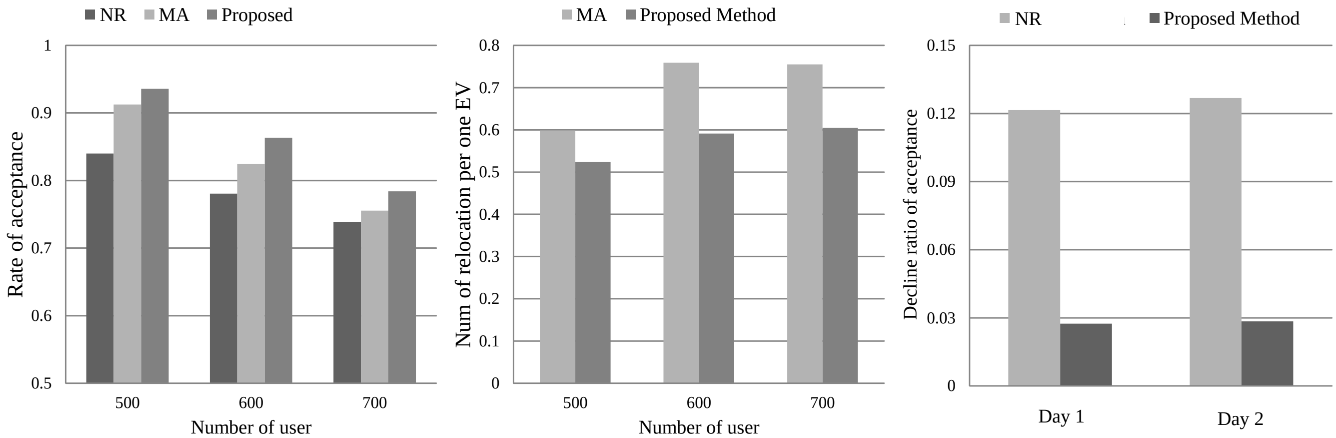
Here, we show example schedules for a user and an EV, obtained by the proposed method. Table III shows the accepted reservation for the user. ID 12 is assigned to this user, and Table IV is the user schedule found for this reservation. The initial station for this user is 27, and he/she departs from this station with EV 131. He/she arrives at station 5 at 8:44. After 78 minutes, which is the specified stay time for this station, he/she departs from this station with EV 131, and arrives at station 39 at 10:34. Since the charging of EV131 is not finished by the departure time, he/she uses EV 191 instead to move to the final destination. Table V shows the corresponding schedule for EV 131. EV 131 is first driven by user 12. It departs from station 27 and arrives at station 5. At station 39, it is recharged, after which it is ready for a new reservation.

TABLE IV
EXAMPLE OF USER SCHEDULE

Time	Action
08:06	Depart from station 27(EV131)
08:44	Arrive at station 5
	Stay at station 5
10:02	Depart from station 5(EV131)
10:34	Arrive at station 39
	Start charging EV131
	Stay at station 39
11:49	Depart from station 39(EV191)
11:58	Arrive at station 57
	Stay at station 57
13:10	Depart from station 57(EV191)
13:29	Arrive at station 43

D. Acceptance rate

Fig. 4(a) shows acceptance rates of reservation requests with the three methods for 500 to 700 users. The result shows that



(a) (b) (c)

Fig. 4. Simulation results (a) Acceptance rate (b) Relocation count (c) Decline of acceptance rate

TABLE V
EXAMPLE OF EV SCHEDULE

Time	Action
08:06	Depart from station 27 (User 12, battery 16.0)
08:44	Arrive at station 5 (User 12, battery 11.4)
10:02	Depart from station 5 (User 12, battery 11.4)
10:34	Arrive at station 39 (User 12, battery 7.6)
	Start charging at station 39
12:34	Standing by at station 39 (battery 15.6)

in each case the acceptance rates are highest with the proposed method. The proposed method improved the acceptance rate by 4% compared to the MA method.

E. Relocation count

Fig. 4(b) shows the result in comparing the relocation counts of the MA method and the proposed method as the number of users is changed from 500 to 700. In each case the proposed method relocates fewer EVs than the MA method. Also, when the number of users increases, the increase in relocation counts is least with the proposed method. This is because the proposed method gives priority to relocating EVs to neighboring, after which it tries to lower the difference in margins at each station.

F. Decline of acceptance rate associated with congestion

In order to assess how congestion influences the acceptance rate, we compared the ratio of decline in acceptance rates between simulation scenarios with and without congestion. We compared the acceptance rates under the congested and non-congested scenarios with the same users making the same reservation requests at the same timing. We used traffic obtained from JARTIC for two different days. Fig. 4(c) shows the decline ratios of acceptance with 700 users for each day. The decline ratio is 3% for the proposed method and 12% when EVs are not relocated. The results indicate that the relocation in the proposed method works to reduce the number of rejected reservation requests. This is due to maintaining a margin of EVs at each station and thereby reducing the influence of delayed EV arrivals.

VII. CONCLUSION

In this paper, we proposed an online scheduling system for many users who visit multiple destinations by means of one-way electric vehicle sharing. The proposed system enables users to continue their journeys without waiting for recharging at each station. In our simulation-based evaluation with 340 EVs and 68 stations, 78% of 700 reservation requests are accepted. This means that 1.6 users are riding each EV in average, and thus EVs are more efficiently utilized than the case where each user owns an EV.

REFERENCES

[1] Autolib official web site. autolib. [Online]. Available: https://www.autolib.eu/en/

[2] car2go official web site. car2go. [Online]. Available: https://www.car2go.com/en/austin/

[3] M. Barth and M. Todd, "Simulation model performance analysis of a multiple station shared vehicle system," *Transportation Research Part C: Emerging Technologies*, vol. 7, no. 4, pp. 237–259, 1999.

[4] M. Barth, M. Todd, and L. Xue, "User-based vehicle relocation techniques for multiple-station shared-use vehicle systems," in *Transportation Research Board 80th Annual Meeting Jan. 2004 Washington, D.C.*, 2004.

[5] A. D. Febbraro, N. Sacco, and M. Saeednia, "One-way carsharing: Solving the relocation problem," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 4, no. 2319, pp. 113–120, 2012.

[6] A. Kek, R. Cheu, and M. Chor, "Relocation simulation model for multiple-station shared-use vehicle systems," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1986, pp. 81–88, 2006.

[7] S. Smith, M. Pavone, M. Schwager, E. Frazzoli, and D. Rus, "Rebalancing the rebalancers: optimally routing vehicles and drivers in mobility-on-demand systems," *American Control Conference (ACC), 2013*, pp. 2362–2367, 2013.

[8] M. Bruglieri, A. Colorni, and L. Alessandro, "The relocation problem for the one-way electric vehicle sharing," *Networks*, vol. 64, no. 4, pp. 292–305, 2014.

[9] Japan road traffic information center. Japan Road Traffic Information Center. [Online]. Available: http://www.jartic.or.jp/