IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 13 December 2023, accepted 4 February 2024, date of publication 13 February 2024, date of current version 23 February 2024.

*Digital Object Identifier 10.1109/ACCESS.2024.3365947*

RESEARCH ARTICLE

# Optimizing Shared E-Scooter Operations Under Demand Uncertainty: A Framework Integrating Machine Learning and Optimization Techniques

**NARITH SAUM[1,2], SATOSHI SUGIURA[1], AND MONGKUT PIANTANAKULCHAI[2]**

[1]Division of Engineering and Policy for Sustainable Environment, Hokkaido University, Sapporo, Hokkaido 060-0808, Japan
[2]School of Civil Engineering and Technology, Sirindhorn International Institute of Technology, Thammasat University, Pathum Thani 12121, Thailand

Corresponding author: Narith Saum (saumnarith@gmail.com)

This work was supported in part by the ASEAN University Network/Southeast Asia Engineering Education Development Network (AUN/SEED-Net) Collaborative Education Program (CEP) through the Sirindhorn International Institute of Technology, Thammasat University, and Hokkaido University; in part by Japan Society for the Promotion of Science (JSPS) KAKENHI, under Grant 22H01610; and in part by the Committee on Advanced Road Technology under the authority of the Ministry of Land, Infrastructure, Transport, and Tourism in Japan, through the Project "Research on the Evaluation of Spatial Economic Impacts of Building Bus Termini," Principal Investigator: Prof. Yuki Takayama, Kanazawa University.

**ABSTRACT** The emergence of dockless shared e-scooters as a new form of shared micromobility offers a viable solution to specific urban transportation problems, including the first-mile–last-mile issue, parking constraints, and environmental emissions. However, this sharing service faces several challenges in daily operation, particularly related to demand volatility, battery recharging, maintenance, and regulations, owing to their trip and physical characteristics. Therefore, this study proposed a new data-driven rebalancing framework for dockless shared e-scooters that incorporates demand and variance prediction, and Monte Carlo sampling to simulate the expected demand. Thus, demand uncertainty and the collection of low-battery and broken e-scooters were included in the rebalancing formulation to minimize user dissatisfaction and operating costs. Rebalancing optimization is an NP-hard problem; in this study, the small-size problem was solved using the integer linear programming (ILP) solver GNU Linear Programming Kit, and the large-size problem was solved using the proposed hybrid ant colony optimization–ILP algorithm (ACO–ILP). This framework was evaluated on a real-world dataset from Minneapolis, Minnesota, which demonstrated that the demand and variance prediction efficiently allocated the uncertainty while reducing the overall uncertainty, leading to shorter driving distances and lower rebalancing costs relative to baseline cases.

**INDEX TERMS** Shared e-scooter, integer linear programming, ant colony optimization, gradient boosting, SGARCH, demand uncertainty.

## I. INTRODUCTION

The first-mile–last-mile problem is a common transportation problem in many urban areas globally, owing to improper planning of public transit, in addition to budget constraints and urban sprawl. The private sector has stepped in to fill this transportation gap by supplying shared transportation modes—shared bikes, shared electric (e-)bikes, and shared e-scooters—that are operated as either dock-based or dockless systems. Dock-based systems are mainly used for shared bikes and require bikes to be picked up or dropped off at specific stations. As the capacity of a station is dictated by its number of docks, users of shared dock-based bikes sometimes cannot finish their trips at the most convenient station if it has no available capacity and thus have to finish their trips at a less convenient station that has available capacity. In contrast, users of dockless shared bikes or e-scooters can pick up or properly park a dockless shared bike or e-scooter at any public space within a given operational area. However, although this dockless shared mobility is convenient, dockless bikes or e-scooters sometimes obstruct the public (such as by impeding sidewalk access), negatively

The associate editor coordinating the review of this manuscript and approving it for publication was Wai-Keung Fung.

affect urban esthetics, or are vandalized. To handle these problems, operators are required to remove excess vehicles in a timely manner or provide proper parking areas. Another type of distribution regulation involves the necessity of distributing shared e-scooters to disadvantaged communities, such as those with low income, minority populations, or limited access to public transit, in order to ensure vehicle equity [1], [2].

Shared bikes were first introduced in Amsterdam in 1965 [3], whereas shared e-scooters were first introduced in Singapore in 2016 and in the United States in 2017 [1]. Nevertheless, in the United States in 2019, the total number of trips of shared e-scooters (96 million trips) surpassed that of dockless and dock-based shared bikes combined (40 million trips), as e-scooter services are provided in more than 100 cities in the United States [4]. This shared micromobility has been comprehensively reviewed [1], [5], [6]. Moreover, shared e-scooters have gained much attention from researchers in terms of policy and regulation [2], [7], [8], [9], [10], [11], [12], spatiotemporal trip characteristics [5], [13], [14], [15], [16], [17], [18], life cycle assessment [19], [20], [21], [22], and social perception [23], [24], [25], [26].

Regarding short-term demand prediction, a novel spatiotemporal graph capsule neural network called GCScoot has been developed to forecast the trip flow of shared e-scooters by considering deployment reconfiguration [27], [28]. GCScoot achieved start-of-art performance compared with baseline models when it was evaluated on open datasets from four cities in the United States (Austin, Texas; Louisville, Kentucky; Minneapolis, Minnesota (MN); and Chicago, Illinois). In addition, other recent studies have employed or proposed various prediction models, such as seasonal autoregressive integrated moving average and generalized autoregressive conditional heteroskedasticity (GARCH) [29], encoder–recurrent neural network–decoder framework [30], masked fully convolutional networks [31], graph convolutional networks [32], long short-term memory [33], and a bagging ensemble of several decision tree-based models [34].

However, there have been few studies on the operational planning of shared e-scooters, including recharging e-scooters, fleet size design, e-scooter distribution and rebalancing, and facility location design. Masoud et al. [35] addressed a shared e-scooter recharging problem formulated as an ILP by modifying a College Admission (CA) algorithm to determine the optimal allocation of freelance chargers. Similarly, Ciociola et al. [36] adopted Poisson processes to examine the effects of fleet size and battery charging on the simulated demand of shared e-scooters. Additionally, the deep learning model 3D-CLoST was used to predict shared e-scooter demand before applying a greedy relocating strategy by workers [37]. Osorio et al. [38] incorporated the possibility of charging e-scooters on overnight rebalancing vehicles and formulated a mixed-integer program. To solve large-instance problems, they developed a discrete-continuous hybrid model that integrates line haul and local operations and evaluated the model using a randomly generated demand based on normal distributions. Fathabad et al. [39] developed a two-stage stochastic program for short- and long-term operational planning of shared e-scooters. The first stage minimizes the investment costs (of charging facilities, e-scooter fleet size, and relocating schedules), whereas the second stage optimizes the short-term operational costs (the relocation cost, charging cost, and penalty of unserved demand). Losapio et al. [40] devised E-Scooter Balancing–Deep Q Network, a multi-agent deep reinforcement learning approach that minimizes a rebalancing operation and battery swapping by incentivizing customers to pick up e-scooters in one-hop neighboring zones. The location of charging stations for shared e-scooters was optimized, taking into account energy consumption influenced by dynamic motion, location model, 3-dimensional road geometry, and road surface characteristics [41]. Finally, [42] optimized the locations of charging stations for a shared e-scooter system by using a multi-criteria decision protocol based on a geographical information system, with the aim of integrating the shared e-scooter system with existing public facilities, points of interest, and population densities.

Shared e-scooters are mainly used for short trips (i.e., a distance and duration of approximately 1.5 km and 10 minutes, respectively) and particularly for tourism and recreational activities [6], [29], [31], [43], and their ridership is more volatile than that of shared bikes, which are mainly used for commuting. Shared bike ridership exhibits two peak-demand periods, one during the morning rush hour and one during the evening rush hour, whereas shared e-scooter ridership is high from morning until late evening. The pickup and drop-off demand in a dock-based system is bounded by the number of docks or station capacity, whereas the dockless system does not have demand boundaries, resulting in greater fluctuations in demand or higher demand volatility. Owing to shared e-scooters' ridership pattern and dockless nature, they require more frequent rebalancing to satisfy their fluctuating demand. In other words, two rebalancing operations for the two peak demand periods of shared bikes may be sufficient, but shared e-scooters may need a greater number of rebalancing operations and a shorter planning horizon. Moreover, shared e-scooters require intensive maintenance [16] and have a short service life [20], as they are built to be lightweight and easy to ride, and require battery swap-out or recharging when their battery power is low. Compared with the former scenario, the latter scenario (i.e., recharging the battery) is easier to incorporate with the rebalancing process and thus was the focus of the current study. In this scenario, e-scooters with low-battery power are relocated to nearby charging facilities, particularly charging stations relying on renewable energy, such as solar power [44].

Few previous studies on shared e-scooters have investigated short-term operational planning (i.e., within one

day planning such as hourly, etc.), particularly regarding accounting for demand uncertainty. In this regard, a queuing model with Poisson distribution of demand is commonly employed in bike sharing, but this technique has a few disadvantages. First, the demand uncertainty in the queuing model is higher compared with the demand prediction approach, and covering this uncertainty results in high operational costs (see Fig. 2). Second, the actual demand of shared micromobility is highly volatile and affected by many exogenous factors, so it does not follow Poisson distribution (see Section IV-A). On the other hand, several studies have applied machine learning or deep learning models to predict short-term demand for rebalancing operations. However, deploying e-scooters solely based on predictions from their models results in an undesired service level because prediction error is not considered. Therefore, the present study developed a novel data-driven framework for the short-term rebalancing of shared e-scooters that considers both e-scooter and trip characteristics. The main objective of this framework is to relocate the limited number of e-scooters to areas projected to have the highest expected demand. The uncertain demand of shared e-scooters is primarily minimized by demand prediction, whereas the variance prediction model, SGARCH, allocates the remaining uncertainty based on the conditional heteroskedasticity approach. In this context, the term ''allocation'' refers to the time dependence of forecasted variance by SGARCH on the current trend of demand volatility and the performance of demand prediction models compared to constant or seasonal (daily or weekly) variances. Our framework is well-suited for static rebalancing planning within a planning horizon ranging from a few to a couple of hours. However, the proposed framework can also be expanded to accommodate multiple planning horizons in future research. The contributions of this study are as follows.

- Monte Carlo sampling was adopted to simulate the demand uncertainty of shared e-scooters according to the trip gap predicted by Gradient Boosting (GB) regression and the variance and probability distribution predicted by the seasonal generalized autoregressive conditional heteroskedasticity (SGARCH).
- The static vehicle-based rebalancing problem was formulated as an ILP problem to address demand uncertainty, the collection of broken e-scooters to the depot, the relocation of low-battery e-scooters to nearby charging stations, and distribution regulations. Two ILP formulations were constructed with known and unknown route sequences, while the objective function and relevant constraints were modified to facilitate practical implementation, specifically by penalizing specific unmet demands rather than deviations in requests.
- Integer Linear Programming solver (GLPK) and a hybrid ant colony optimization–ILP (ACO–ILP) algorithm were used and proposed, respectively, to solve the rebalancing tasks with demand scenarios generated by the Monte Carlo approach. A real-world dataset of dockless shared e-scooters operating in Minneapolis (MN) was chosen as the case study.

The remainder of this paper is separated into five sections. Section II reviews studies on the rebalancing of transportation-sharing services, especially bike sharing. Section III presents the methodology used in the research framework, demand prediction, variance prediction, rebalancing formulation, and the ACO–ILP algorithm. Section IV presents the data collection, description, and predicted short-term demand and variance. Section V provides the empirical results of rebalancing optimization, and Section VI concludes this paper and suggests areas for future investigation.

## II. LITERATURE REVIEW

### A. REBALANCING OF SHARING SERVICES

As mentioned in the previous section, there has been limited research on the operational planning of shared e-scooters, but we can also learn from other sharing services, especially shared bikes, which are the most similar modes to shared e-scooters. Shui and Szeto [45] reviewed studies and found that they have focused on different characteristics of shared-bike rebalancing, such as objective functions (e.g., distance, cost, and emission), constraints (e.g., budget, service time, and inventory), optimization algorithms (exact or heuristic algorithms), deterministic or stochastic problems, and static or dynamic problems. Several other studies also summarized these challenges of rebalancing problems, including methodology [46], objectives [47], [48], problem size [49], number of rebalancing vehicles [50], [51], damaged bikes [52], equilibrium of station [53], and multi-step matching [54]. The present study focused on problems of rebalancing shared bikes under demand uncertainty and with predicted demand.

Shared bikes are commonly used for commuting and thus have a peak-demand period in the morning (6 am–10 am) and in the evening (4 pm–8 pm) [47], [55]. Owing to this and their stations' capacities, the demand for shared bikes is more stable than that for dockless shared e-scooters. Therefore, studies have typically assumed that bike-sharing demand follows a Poisson distribution. For instance, it has been assumed that a dynamic shared-bike inventory level can be represented using continuous-time Markov chains (CTMCs) with Poisson processes of pickups and drop-offs, and thus this level has been modeled as a double-ended queuing system. The results of this modeling approach were examined using a real-world dataset, bike-sharing in Tel Avis, Israel [56]. Similarly, the unsatisfied demand or service level was derived from the historical data of bike sharing in Palma de Mallorca, Spain, using a simulation approach (Monte Carlo approximation) and an approximate approach (Skellam distribution) [57]. The approach of using CTMCs was adapted for the overnight rebalancing of the Citi Bike system in New York City, whereas small- and large-scale problems have been solved using the ILP solver and a greedy algorithm, respectively [58]. Likewise, a non-stationary queuing (Mt/Mt/1/K) model with exponentially distributed

pickups and drop-offs was developed for the static rebalancing of bike-sharing services in Boston, Massachusetts (Hubway), and Washington, DC (Capital Bikeshare) [59]. Seo [60] accounted for the demand uncertainty in dynamic rebalancing by adopting a Markov decision process based on Poisson distribution, whose mean was the demand predicted by a random forest regression. The author examined bike sharing in Seoul, South Korea, as a case study, but a chi-squared goodness-of-fit test showed that only 77% of the stations in the study had demands that followed a Poisson distribution. Lu [61] developed a robust multi-period bike fleet allocation scheme by assuring the worst-case scenario (i.e., maximum demand) for bike sharing in New Taipei City, Taiwan.

Other studies have accounted for demand uncertainty in rebalancing problems using sample average approximation from Monte Carlo sampling. For example, the operational planning of shared autonomous electric vehicles in Shanghai, China, was conducted using the Monte Carlo method to generate daily demand from a normal distribution based on the average demand [62]. Regarding bike sharing, Monte Carlo sampling was used to generate demand scenarios in Bergamo, Italy, based on four probability distributions (uniform, exponential, normal, and log-normal probability distributions) with a mean and standard deviation derived from historical data [63]. Similarly, demand scenarios for New Taipei City, Taiwan, were generated from a truncated normal distribution with a mean and standard deviation derived from historical demand (weekly) data [64]. Instead of using the aforementioned sampling technique, Dell'Amico et al. [65] used historical data in each day as scenarios and solved stochastic programming models via branch-and-cut, deterministic equivalent program, L-shaped, and heuristic algorithms based on several open datasets.

On the other hand, machine learning and deep learning models have satisfactory prediction performance and have thus been integrated into rebalancing frameworks. For example, Regue and Recker [66] formulated chance-constrained programming for dynamic bike-sharing rebalancing based on a normal distribution, which had the demand predicted by GB and the model error on a test set as its mean and variance, respectively. A random forest regressor was used to forecast the station-level rental and return demands of bike sharing in Nanjing, China, with static rebalancing formulated according to predicted demand and as a hub-first–route-second problem [67]. A spatiotemporal graph neural network was devised to predict city-wide bike demand for truck-based rebalancing in New York City [68]. Similarly, a deep learning model was embedded in a data-driven framework for a deterministic dynamic rebalancing of bike sharing in Beijing, China [47]. A random forest was used to forecast future demand and inventories for the repositioning of bike sharing in Seoul, South Korea [69]. Yu et al. [70] constructed a hybrid model, SARIMA-LSTM, to predict pickup and drop-off demands, which they used to plan rebalancing for bike sharing around rail transit stations in Xicheng, Beijing.

In summary, the methods for the short-term operational planning of bike sharing have mostly accounted for demand uncertainty by using a Markov chain and assuming that demand follows a probability distribution (usually a Poisson distribution). Otherwise, rebalancing has been planned using demand predicted by machine learning and deep learning models. Regue and Recker [66] included the error of the demand prediction model in their rebalancing of shared bikes, but they did not properly examine the variance and probability distribution. Therefore, the present study used SGARCH model to examine the residuals of the demand prediction model, as this regression model can reduce the average demand uncertainty and provides the probability distribution and temporal variance, which are the essential parameters for Monte Carlo sampling. A few previous studies have included collecting broken bikes in their rebalancing problems [47]. However, no studies have particularly addressed the distribution regulations and the recharging issue, specifically for electric bikes (e-bikes). This lack of attention can be linked to the fact that, despite their high cost, similar to shared e-scooters, e-bikes are frequently deployed as dock-based mode at low percentages alongside regular bikes. Nonetheless, e-bikes are generally perceived as less attractive. Hence, this study aims to handle all four characteristics of dockless shared e-scooters: normal/usable, faulty/broken, and low-battery e-scooters, and distribution regulations. Furthermore, this study formulates the rebalancing problem with stochastic demand based on Sample Average Approximation (SAA) approach, so that a target service level can be achieved through parameter settings. This approach also allows us to formulate the rebalancing task as an ILP problem with unknown and known route sequences, which can be solved using exact and heuristic algorithms, respectively.

### B. OPTIMIZATION ALGORITHMS

Rebalancing problems in shared micromobility, including e-scooter and bike sharing, involve the use of dedicated rebalancing vehicles and can be classified as a variant of the one-commodity pickup and delivery capacitated Vehicle Routing Problem (VRP) or one-commodity pickup and delivery Traveling Salesman Problem (TSP). When dealing with multiple types of e-scooters or bikes, these problems become multi-commodity VRP or TSP [71], which is also the focus of this study. Therefore, these rebalancing problems fall into the category of NP-hard optimization problems, where computational time grows exponentially with the number of nodes. These rebalancing problems are typically addressed using exact or heuristic algorithms, depending on problem size. For instance, rebalancing problems in shared micromobility are often formulated as integer programming, allowing them to be tackled by exact algorithms such as (mixed) ILP (with decomposition) [54], [66], [72], [73], branch-and-cut [55], [74], branch-and-bound [75], branch-and-reject [76], and constraint programming (CP) [59].

Various heuristic algorithms have also been developed or applied to address these NP-hard optimization problems. These include (hybrid) genetic algorithm [48], [71], ant colony optimization with CP (ACO–CP) [77], discrete-continuous hybrid model [38], CA [35], extended particle swarm optimization [52], [53], greedy-genetic heuristic [46], tabu search [49], neighborhood search [50], and neighborhood search–variable neighborhood descent [51]. Other algorithms can be found tabulated in the references [46], [47], [48], [50].

NP-hard optimization problems often pose challenges for ILP solvers in generating desirable solutions, or even feasible ones, within limited time constraints, particularly in the case of stochastic problems. Conversely, heuristic algorithms, while unable to guarantee the global optimal solution, often provide better feasible solutions within a restricted computational time frame. To ensure scalability, this study proposed a hybrid heuristic algorithm, named ACO–ILP algorithm, which combines ant colony optimization (ACO) with an ILP solver to solve the rebalancing optimization tasks of dockless shared e-scooters. A similar hybrid algorithm, ACO–CP, was developed for the deterministic rebalancing of bike sharing [77]. However, in our study, ACO was utilized to generate a population of route sequences in each iteration, while the ILP solver was employed to optimize the pickup and drop-off operations specifically for each of these predefined route sequences (i.e., rebalancing with a known route sequence). The proposed ACO–ILP algorithm also supports parallel computing, which could substantially reduce computational time, especially when dealing with large-scale problems. This feature makes it suitable for scalability and practical implementation.

## III. METHODOLOGY

### A. RESEARCH FRAMEWORK

The typical procedure for dockless shared e-scooter trips involves several steps: searching for nearby e-scooters by looking around or using a phone application, walking to the e-scooter location, unlocking the e-scooter with a phone application, riding the e-scooter to the destination, properly parking the e-scooter, and finally, terminating the trip. Although the trip termination process in dockless mode is very convenient, users may choose not to start their trip with an e-scooter if the distance they have to walk to pick up the e-scooter is too far. To address this issue, operators commonly partition the operational area into walkable (ex., 200 – 500 meters) zones and ensure that e-scooters are always present in these zones. One common approach operators employ is rebalancing or relocating e-scooters from zones with excess e-scooters to zones with fewer e-scooters. Under various operational constraints, especially the limited availability of e-scooters, operators regularly assess the status of each zone and perform rebalancing activities to minimize unsatisfied demands (or staving zones) during the planned timeframe. To improve operational efficiency, operators can utilize relevant information and historical trip data to forecast future demands, i.e., the number of pickup and drop-off demands in each zone during specific time intervals. This information is then utilized to optimize the rebalancing task, which involves determining the route path for the rebalancing vehicles and the number of e-scooters to be picked up or dropped off in each zone.

Fig. 1 shows the research framework, which consisted of three main parts: (1) data collection and manipulation, (2) trip gap prediction using GB regression and variance prediction using SGARCH, and (3) rebalancing optimization. As shared e-scooters are a short-range transportation mode, their short-term demand is susceptible to external factors, such as weather, seasonal (weekly and yearly) factors, holidays, and special events [5]. These factors were thus considered and manipulated in the demand prediction models. E-scooters are a dockless or free-floating shared-transportation mode, so users can pick up and park e-scooters anywhere within a given operating area, except on private properties or in forbidden zones stipulated by authorities. Gridding is commonly used to aggregate spatial trips, but this technique does not account for the concentration of demand or the similarity of trip purposes. For example, a group of trips in a shopping mall, park, or school may be separated into a few cells if the group is located on the edge of a grid. Additionally, studies have observed aggregation based on postal codes or administrative areas (communities or wards), but these areas are often too large for the short-term rebalancing of shared e-scooters. Hence, the present study used a $k$-means clustering algorithm to aggregate the spatial ridership of shared e-scooters for the number of clusters of 15, 30, and 60.

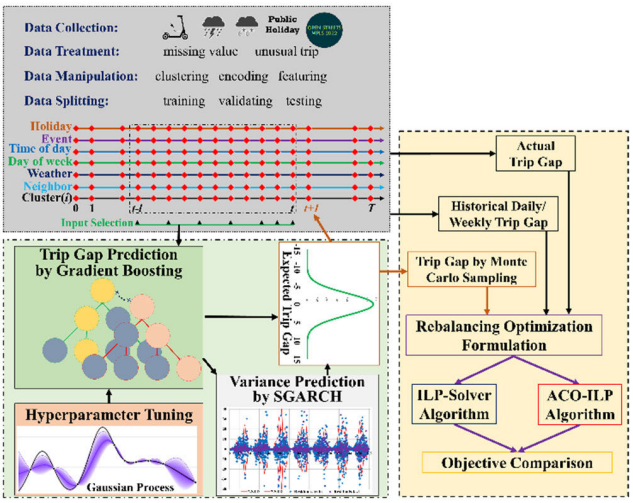


**FIGURE 1.** Research framework.

To deal with the sparsity of trip flow, the present study predicted the trip gap or net demand, which is the difference in demand between the trip-starts (i.e., the trip generation or pickup demand) and trip-ends (i.e., the trip arrival or drop-off demand). A positive trip gap indicates that there are more trip-starts than trip-ends. The hyperparameters of

the GB regressor for trip gap prediction were optimized through Bayesian Optimization (BO), and the residuals of trip gap prediction were used to train the SGARCH model. Then, based on the predicted trip gap and variance, Monte Carlo sampling was conducted to simulate the demand uncertainty of shared e-scooters for rebalancing optimization. This optimization problem was solved using the ILP solver GNU Linear Programming Kit (GLPK) and the hybrid ACO–ILP algorithm. Studies have typically derived the stochastic nature of shared-bike demand from historical data, where each scenario has been a seasonal instance, or a sample obtained by Monte Carlo simulation [63], [64], [65]. To avoid an assumption about the distribution of demand (as this is not a practical assumption regarding the demand for shared e-scooters), seasonal (daily and weekly) historical data were selected as baseline cases for rebalancing planning. Finally, the objective values for all the rebalancing optimization problems—the ILP solver and ACO–ILP results for the actual trip gap, the historical daily and weekly trip gap, and the simulated trip gap—were compared. The comparison was examined using 30 random instances from the testing dataset (see Section V). The aim of comparing the ILP solver with the proposed ACO–ILP algorithm is to demonstrate the effectiveness of these two algorithms in handling different problem sizes, thus highlighting their scalability. Additionally, the purpose of comparing the simulated demands by the Monte Carlo method with the baseline cases, namely historical daily and weekly trip gaps, is to showcase the effectiveness of minimizing and allocating the demand uncertainty through demand and variance prediction.

### B. DESCRIPTION OF REBALANCING PROBLEM

Due to data limitations and the complex nature of actual operations of shared micromobility involving bikes, e-bikes, or e-scooters, operational planning is usually based on several assumptions that may vary between studies. Even though some assumptions can be allocated through parameter setting, worst-case scenario, etc. The present study made the following assumptions.

- *Assumption 1:* The distributions of the trip-start (pickup) and trip-end (drop-off) of a specific cluster are uniform along the time interval ($\Delta t$). That is, all of the e-scooters from the trip-end can be used for pickup trips, particularly if the drop-off demand is lower than the pickup demand (or the trip gap is positive). For instance, an empty cluster (i.e., one with no e-scooter in its area) that has 15 trip-starts and 10 trip-ends (i.e., a trip gap of +5) is supposed to have only five unmet demands if there is no other interruption, whereas it could have up to 15 unmet demands if all of the trip-starts are in the first half of the time interval and all of the trip-ends are in the second half of the time interval.
- *Assumption 2:* The demand in each cluster does not change during the planning and rebalancing process, i.e., static rebalancing planning.
- *Assumption 3:* Owing to the dockless sharing mode, a customer picks up an e-scooter if it is available in the same cluster (i.e., the walking distance is ignored); otherwise, the customer leaves the system (i.e., there is unmet demand).
- *Assumption 4:* Broken e-scooters are repaired at the only one depot, whereas low-battery e-scooters are recharged at charging stations or the depot.
- *Assumption 5:* There is only one rebalancing vehicle, and it must visit all of the nodes, including the charging stations and demand clusters.

The demand uncertainty is mainly minimized in the demand prediction step, commonly presenting in the form of a smaller mean squared error (MSE) compared to the historical average. Since the explanatory features are included in the demand prediction model, GB, the residuals are supposed to be white noise or random walk. However, the remaining uncertainty (or variance of residuals) can be further mitigated through a variance prediction model explicitly designed for heteroscedastic datasets. Even though, the primary objective of variance prediction is to allocate uncertainty based on the concept of conditional variance, i.e., temporal variance. Using the predicted trip gaps and variances, the operator can relocate the limited number of e-scooters to the area that maximizes the expected profit.

This paper addresses the rebalancing problem on a complete graph $G = (N, A)$, where $N$ is the set of all nodes (including depot, charging stations, and demand clusters), and $A$ is the set of links or edges between these nodes. Other notations used in this study are presented in Table 1. Three types of e-scooters are considered: faulty (or broken), low-battery, and usable (or normal) e-scooters. Faulty e-scooters are collected and brought back to the depot for repair, whereas low-battery e-scooters are relocated to charging stations or the depot for battery recharging. Faulty e-scooters refer to those with electronic or frame issues that require repair by technicians at the depot, and this status is commonly reported by customers. The operator has the flexibility to define the threshold of battery level for categorizing e-scooters as low-battery (e.g., the battery level required for an average trip duration or the entire planning horizon). Therefore, the status of these two types of e-scooters is assumed to be known by the operator during the planning stage. In this case, several parameters represent the current information known at the beginning of the planning horizon, such as vehicle capacity ($B$), driving distance ($c_{i,j}$), the number of faulty ($v_i^f$), low-battery ($v_i^l$), and usable ($v_i^u$) e-scooters at each node, as well as the number of charging docks ($D_i$).

The expected trip gap in this study is assumed to be a parameter for rebalancing optimization despite being simulated based on the predicted trip gap and variance. The demand at the depot and charging stations is assumed to be zero, while the expected net demand $g_i^\theta$ in each cluster $i$ is simulated for the total scenarios of $\Theta$ using Monte Carlo approximation. Several other parameters are user-defined, including the threshold values for the minimum ($\underline{C}_i$) and

**TABLE 1. List of notations.**

| Notation | | Description |
|---|---|---|
| Set | $N$ | Set of nodes (depot, charging stations, and clusters), with component $n$ |
| | $A$ | Set of links in the network, with component $(i,j)$ |
| | $\Theta$ | Set of scenarios, with component $\theta$ |
| Parameter | $v_i^f$ | Number of faulty e-scooters at node $i$ |
| | $v_i^l$ | Number of low-battery e-scooters at node $i$ |
| | $v_i^u$ | Number of usable e-scooters at node $i$ |
| | $D_i$ | Number of charging docks at node $i$ |
| | $\overline{C}_i$ | Maximum number of e-scooters at node $i$ |
| | $\underline{C}_i$ | Minimum number of usable e-scooters at node $i$ |
| | $g_i^\theta$ | Trip gap in scenario $\theta$ and node $i$ |
| | $B$ | Capacity of vehicle |
| | $c_{ij}$ | Driving distance between nodes $i$ and $j$ |
| | $\beta_0$ | Unit cost of driving distance |
| | $\beta_1, \beta_2, \beta_3, \beta_4, \beta_5$ | Unit costs of penalty functions of picking up e-scooters, remaining faulty e-scooters, remaining low-battery e-scooters, unmet demand, and excess e-scooters, respectively |
| Variable | $R_i^f$ | Remaining faulty e-scooters at node $i$ |
| | $R_i^l$ | Remaining low-battery e-scooters at node $i$ |
| | $h_i^f$ | Number of faulty e-scooters on the vehicle at node $i$ |
| | $h_i^l$ | Number of low-battery e-scooters on the vehicle at node $i$ |
| | $h_i^u$ | Number of usable e-scooters on the vehicle at node $i$ |
| | $U_i^\theta$ | Unmet demand in scenario $\theta$ and at node $i$ |
| | $E_i^\theta$ | Excess e-scooters in scenario $\theta$ and at node $i$ |
| Decision Variable | $x_{ij}$ | Binary: 1 if the rebalancing vehicle passes the link $(i,j)$, 0 otherwise. |
| | $a_i$ | Nonnegative-integer: auxiliary variable for subtour elimination |
| | $p_i^f$ | Nonnegative-integer: number of faulty e-scooters picked up at node $i$ |
| | $p_i^l$ | Nonnegative-integer: number of low-battery e-scooters picked up at node $i$ |
| | $d_i^l$ | Nonnegative-integer: number of low-battery e-scooters dropped off at node $i$ |
| | $p_i^u$ | Nonnegative-integer: number of usable e-scooters picked up at node $i$ |
| | $d_i^u$ | Nonnegative-integer: number of usable e-scooters dropped off at node $i$ |

maximum ($\overline{C}_i$) number of e-scooters, and the unit cost of driving distance ($\beta_0$) and penalty terms ($\beta_1$, $\beta_2$, $\beta_3$, $\beta_4$, and $\beta_5$). The objective of rebalancing planning is to ensure that the number of usable e-scooters remains within the specified thresholds, $[\underline{C}_i, \overline{C}_i]$, which are enforced as optimization constraints. This study introduces the parameter, the minimum number of usable e-scooters ($\underline{C}_i$), which serves as a safety stock, allowing the operator to address the limitations of assumptions (1)–(3), distribution regulations, and potential demand (specifically when the demand prediction model is trained on historical ridership data). Similarly, there are cases where regulations require a timely response to locations with an excessive number of e-scooters. Therefore, this study also introduces the maximum number of e-scooters ($\overline{C}_i$) to prevent such unfavorable situations. However, operators have the flexibility to ignore this parameter by assigning a large value or setting the unit cost of the penalty term ($\beta_5$) to zero. The unit cost of pickup ($\beta_1$) is introduced for two reasons. Firstly, it aims to prevent unnecessary pickups, especially when the pickup and drop-off occur at the same location. Secondly, it serves as a tradeoff for the service level, meaning that demand with a probability smaller than the ratio of $\beta_1/\beta_4$ is considered not worth the cost of pickup. The higher values of $\beta_2$ and $\beta_3$ lead to fewer faulty and low-battery e-scooters remaining in the system, specifically after the rebalancing operation. The unit penalty cost of unmet demand ($\beta_4$) is commonly set proportionally to the revenue or profit, but the higher value might lead to a longer driving distance.

In each cluster $i$ and scenario $\theta$, unmet demand ($U_i^\theta$) occurs when the available ($v_i^u$) and the drop-off amount ($p_i^u$) of usable e-scooters are less than the positive net demand ($g_i^\theta > 0$). Consequently, the total inventory tends to approach the upper bound value of the predicted net demand under the constraint of total usable e-scooters. This approach leads to an improvement in the service level while reducing the impact of potential demand. One capacitated vehicle is assigned to relocate the usable and low-battery e-scooters and collect faulty e-scooters to minimize the operational objective, which is defined as the sum of driving cost, pickup cost, and penalty cost of unmet demand, and the remaining faulty, low-battery, and excess e-scooters in the system.

The rebalancing optimization problem in this study consists of two main types of decision variables: routing variables ($x_{ij}$ and $a_{ij}$), and pickup and drop-off variables for different types of e-scooters ($p_i^f, p_i^l, d_i^l, p_i^u$, and $d_i^u$). Previous studies commonly combined pickup and drop-off operations as a single decision variable, representing pickup or drop-off activities as positive or negative values. In this study, we separate these two activities, ensuring all decision variables are strictly positive integer values (nonnegative-integer). Additionally, we introduce a penalty on pickup activities to minimize unnecessary pickups and to achieve a specific service level. The pickup variable for faulty e-scooters ($p_i^f$) is constrained by the number of faulty e-scooters in each zone. For low-battery e-scooters, the pickup activities ($p_i^l$) may be required if there are low-battery e-scooters present in each demand cluster and if there are more low-battery e-scooters than charging docks at each charging station. The drop-off of low-battery e-scooters ($d_i^l$) is only allowed if there are available charging docks. For usable e-scooters, the number of pickups ($p_i^u$) is allowed if there are more usable e-scooters than the specific safety stock ($\underline{C}_i$), while the drop-offs ($d_i^u$) are constrained by the availability on the rebalancing vehicle.

### C. DEMAND PREDICTION BY GRADIENT BOOSTING

As shown in Fig. 1, the short-term trip gap of shared e-scooters was predicted by GB, a powerful machine-learning

technique proposed by Friedman [78]. A GB model is an ensemble decision-tree model based on a boosting approach and achieves good prediction performance by iteratively adding a new weak learner (decision tree) to minimize remaining errors that have been generated by the previous learners. The base estimator is a classification and regression tree so that GB can be used for both classification and regression problems. The detailed formulation and algorithm of GB were presented by [78], whereas the loop (i.e., iteratively accumulating decision tree) of the GB regressor comprises several essential steps: computation of the negative gradient (starting with the mean value for the initial prediction), fitting of the regression tree to predict the negative gradient, calculation of the gradient descent step size (or learning rate), and updating of the GB model or prediction performance. In this study, the GB model was trained using a Python module in the Scikit-Learn package, *GradientBoostingRegressor* [79].

The GB regressor can achieve performance comparable to deep learning models [80] but requires proper featuring and hyperparameter tuning. There are several approaches for hyperparameter optimization, such as manual search, grid search, random search, sequential model-based approach, and population-based approach. The sequential model-based approach is commonly adopted to optimize hyperparameters of machine learning and deep learning, as it can achieve a near-globally optimal solution within a short computational time. This approach uses surrogate and acquisition functions to iteratively suggest a new candidate until the stopping criteria are reached. Two popular algorithms in this approach are Bayesian Optimization (BO) and Tree-structured Parzen Estimator (TPE). In the BO algorithm, a Gaussian process is used to construct a surrogate function from the evaluated samples (starting with initial random samples), and the new candidate has the highest expected performance (e.g., the minimum mean squared error (MSE) on evaluation data) that can be achieved using the probability of improvement, expected improvement, or lower confidence bound as the acquisition function. Further details of the BO algorithm can be found in the literature [81]. This study selected the BO with lower confidence bound due to its capability to handle local optima, as it has a parameter, *kappa*, that balances the tradeoff between exploration and exploitation.

In the present study, BO with the lower confidence bound was employed to optimize the hyperparameters of the GB algorithm in the Python package Scikit-Optimize or *skopt* [82]. Most parameters of the BO algorithm were set as default values, whereas the number of initial random samples (*n_random_starts*), total number of evaluations (*n_calls*), and coefficient of the lower confidence bound (*kappa*) were set to 50, 200, and 1.8, respectively. BO was adopted to optimize the MSE of the evaluation data and its ratio to that of the training data, i.e., *MSE_eval* + *MSE_eval*/*MSE_train*. This objective function aims to minimize overfitting and reduce the training time by preventing the suggestion of complex models that significantly and slightly reduce *MSE_train* and *MSE_eval*, respectively. This setting of the BO algorithm was applied to optimize five hyperparameters of the GB algorithm: the lookback length ($l$), sampling rate ($r$), number of boosting stages (*n_estimators*), maximum depth of the decision tree (*max_depth*), and learning rate (*learning_rate*). Two of the parameters ($l$ and $r$) were related to input selection (see Fig. 1) to enable the GB regressor to predict the future trip gap ($t + 1$). For hourly prediction, the input selection was drawn from step $t$ to step $t- l - 1$, with one sample being selected in each $r$ interval from $t - l - 1$ to $t - r$, and all of the samples being selected from $t - r - 1$ to $t$. For example, if $l = 24$ and $r = 3$, the list of historical data was $[t - 23, t - 20, t - 17, t - 14, t - 11, t - 8, t - 5, t - 2, t - 1, t]$. The ranges of $l$ and $r$ were [13, 170] and [1, 13], respectively. The lookback length thus covered the weekly pattern for hourly trip gap prediction. The other three parameters (*n_estimators*, *max_depth*, and *learning_rate*) of the GB algorithm were optimized in the ranges [5, 400], [1, 20], and [0.01, 0.5], respectively. Other parameters of the GB algorithm were set as default values; i.e., the loss function was set as the squared error, the minimum required number of samples to split was set as 2, and the split quality was taken as the Friedman MSE.

### D. VARIANCE PREDICTION BY SGARCH

Although machine learning algorithms achieve state-of-art performance, these models still have prediction errors, commonly reported as the MSE, root-mean-square error (RMSE), or mean absolute error. As such, supplying e-scooters by following the demands predicted using these models (i.e., ignoring the error term) would achieve a service level Type I (or probability of shortage events) of only 50%. For example, if operators supply e-scooters to 100 locations based on predicted demands, around 50 locations (right-side of residuals' histogram or probability distribution) may experience a shortage. In other words, the predicted demand is an expected value or mean where roughly 50% of the actual demand lies above or below it. The historical mean and variance have thus been used to simulate stochastic scenarios under an assumed demand distribution. However, the assumed distributions of trip-starts and trip-ends are unrealistic, especially for shared e-scooters, as in

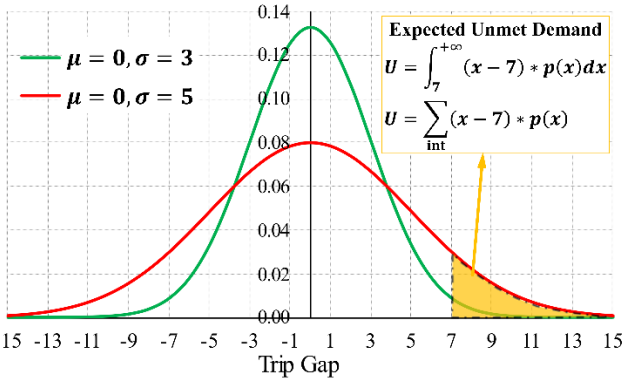


**FIGURE 2.** Effect of demand uncertainty on expected unmet demand.

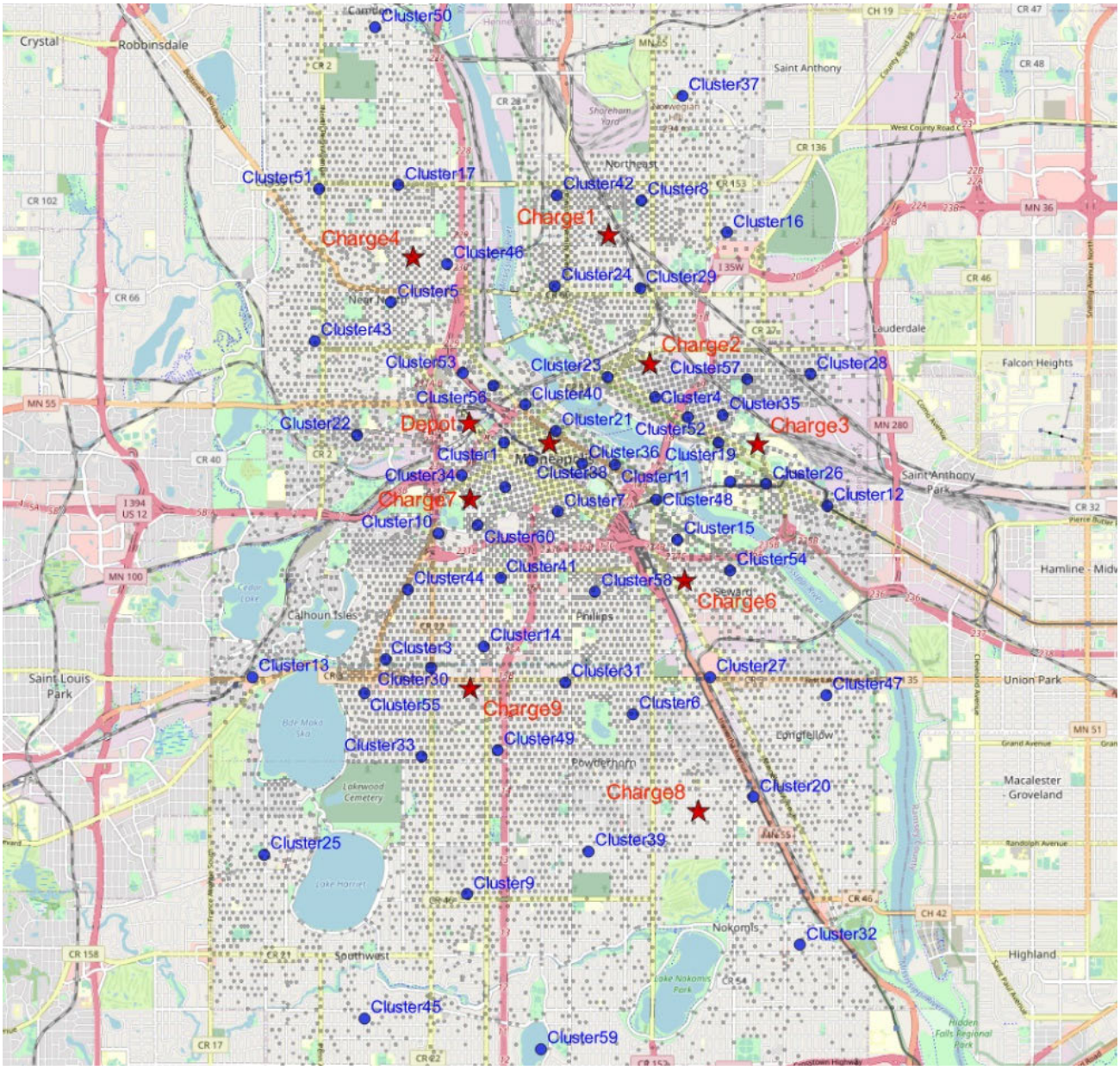


**FIGURE 3.** Trip clustering generated by the *k*-means algorithm (red stars = depot and charging stations; blue dots = centers of trip clusters; gray dots = street centers of pickup and drop-off trips).

Section IV-A, and high uncertainty leads to a high operational cost. For example, as shown in Fig. 2, two Gaussian trip-gap models with different variations give different expected unmet demands; i.e., higher uncertainty results in higher expected unmet demand. Therefore, we can minimize the operating cost by reducing the demand uncertainty through demand prediction and selecting appropriate variation and distribution models.

It was recently found that the residuals of a short-term demand prediction model for shared e-scooters were not white noise [29]. Thus, the present study examined the heteroskedasticity of the residuals of trip gap prediction. As the prediction is a time-series problem, a Lagrange multiplier test was conducted to determine whether the residuals of trip gap prediction from the previous section were homoskedastic; if so, the variance was constant; if not, the conditional variance of the residuals was forecasted using the SGARCH model. An autoregressive conditional heteroskedasticity (ARCH) model can be used to predict future variance on the basis of conditional variance, with high volatility and low volatility grouped together. Such a model has only previous residuals as independent variables, whereas the generalized form (GARCH) also includes previously predicted variances [83]. Therefore, to capture the seasonal pattern, seasonal residuals and predicted variances were added to the GARCH model to construct the SGARCH

model. A comprehensive explanation and other extensions can be found in the reference manual of STATA software [83]. The basic model of SGARCH$(p, q)(P, Q, S)$ is expressed as follows:

$$y_{t+1} = \boldsymbol{X}_{t+1}\alpha + \epsilon_{t+1} \tag{1}$$

$$\begin{aligned} Var\,(\epsilon_{t+1}) = \sigma^2_{t+1} = \gamma_0 &+ \sum\nolimits_{i=1}^{p} \gamma_{p,i}\varepsilon^2_{t+1-i} \\ &+ \sum\nolimits_{i=1}^{q} \gamma_{q,i}\sigma^2_{t+1-i} + \sum\nolimits_{i=1}^{P} \gamma_{P,i}\varepsilon^2_{t+1-iS} \\ &+ \sum\nolimits_{i=1}^{Q} \gamma_{Q,i}\sigma^2_{t+1-iS} \end{aligned} \tag{2}$$

where, $y_{t+1}$ denotes the residuals of the demand prediction model, GB. Therefore, $\boldsymbol{X}_{t+1} = \mathbf{1}$, and $y_{t+1}$ is the sum of a constant value ($\alpha$) commonly close to zero and the disturbance $\epsilon_{t+1}$. In this case, the parameters ($\gamma_0$, $\gamma_{p,i}$, $\gamma_{q,i}$, $\gamma_{P,i}$, and $\gamma_{Q,i}$) of SGARCH model was estimated using the maximum log-likelihood estimator with flexible distributions, i.e., a normal distribution $\epsilon_{t+1} \sim$ N.Dist$(0, \sigma^2_{t+1})$, Student's $T$ distribution $\epsilon_{t+1} \sim$ T.Dist$(0, \sigma^2_{t+1}, \text{df})$, and generalized error distribution. The SGARCH parameters for forecasting the hourly conditional variance $\sigma^2_{t+1}$ were $\sigma^2_t$, $\sigma^2_{t-1}$, $\sigma^2_{t-2}$, $\sigma^2_{t-23}$, $\sigma^2_{t-47}$, $\varepsilon^2_t$, $\varepsilon^2_{t-1}$, $\varepsilon^2_{t-2}$, $\varepsilon^2_{t-23}$ and $\varepsilon^2_{t-47}$ and the insignificant parameters (95%) were removed. This means that the estimated variance at time $t+1$ is strongly influenced by residuals and predicted variances from several of the most recent time steps ($t$, $t$-1, $t$-2) and the corresponding hours from previous days ($t$-23 and $t$-47). This conditional variance estimation properly allocates the uncertainty throughout the day, resulting in a small variance during nighttime and a high variance during daytime. Since SGARCH assigns greater weight to recent trends, it is more tolerant of long-term fluctuations compared to daily variance (i.e., variance at the same hour of the day). The SGARCH model was trained independently for each cluster using the statistical software STATA [83]. Normal and Student's $T$ distributions were considered, and the distribution with the smallest standard deviation was selected. This distribution and forecasted variance were then used to generate the demand uncertainty via Monte Carlo simulation.

### E. REBALANCING FORMULATION FOR THE ILP SOLVER

As reviewed in Section II, previous studies have designed rebalancing problems with various objective functions, predominantly focused on total driving distance [55] and generalized cost [47], [65], [67], [68], [77]. These generalized cost functions often share common terms, such as driving distance or duration, as well as constraints like vehicle capacity, pickup and drop-off constraints, among others. However, there are also distinct terms and constraints specific to each study in order to fulfill their respective objectives. For instance, Chang et al. [47] devised deterministic rebalancing problems for dockless bike sharing, incorporating a generalized cost function encompassing driving cost, pickup and drop-off cost, and penalty cost for unvisited zones where requests exist. In contrast, Dell'Amico et al. [65] formulated stochastic rebalancing for station-based bike sharing, with a generalized cost function comprising driving cost and penalized cost for supply deviation (slack and surplus) from stochastic requests.

In this study, we adopt a generalized cost function as the objective for stochastic rebalancing of dockless shared e-scooters, aiming to minimize the cost associated with driving distance and pickup/drop-off operations. All four main characteristics of this shared service are taken into account, including stochastic demand, low-battery e-scooters, broken e-scooters, and distribution regulations. Instead of imposing penalties for deviation values from the requests, as seen in previous studies, our approach introduces penalties related to unmet demand, excess e-scooters, and remaining faulty (i.e., broken or damaged) and low-battery e-scooters. The penalty for unmet demand $U^\theta_i$ includes both simulated demands in each scenario $g^\theta_i$ and the safety stock inventory $\underline{C}_i$, which may also incorporate a distribution regulation requiring a minimum number of e-scooters in a specific region. Another distribution regulation (i.e., operators need to respond to flooded e-scooters promptly) is addressed by imposing penalties for excess e-scooters when the total e-scooters in a specific zone surpass a specific threshold, $\bar{C}_i$. Consequently, our objective function is more comprehensible to non-experts, particularly when it comes to parameter adjustment for practical implementation. Additionally, this function facilitates the tradeoff between driving distance and pickup and drop-off operations rather than strictly adhering to request constraints.

As shown in the research framework in Fig. 1, the Monte Carlo method was used to generate the demand uncertainty based on the predicted trip gap (Section III-C) and the predicted variance and distribution (Section III-D). These random simulated trip gaps were assumed to have equal probability or weight; therefore, the formulation of the short-term rebalancing of shared e-scooters is as follows.

$$\begin{aligned} \textbf{\textit{Minimize}}\ & \beta_0 \sum\nolimits_{(i,j)\epsilon A} c_{ij}x_{ij} + \beta_1 \sum\nolimits_{i\epsilon N} \left(p^f_i + p^l_i + p^u_i\right) \\ &+ \beta_2 \sum\nolimits_{i\epsilon N} R^f_i + \beta_3 \sum\nolimits_{i\epsilon N} R^l_i \\ &+ \frac{\beta_4}{\Theta} \sum\nolimits_{i\epsilon N;\theta\epsilon\Theta} U^\theta_i + \frac{\beta_5}{\Theta} \sum\nolimits_{i\epsilon N;\theta\epsilon\Theta} E^\theta_i \end{aligned} \tag{3}$$

$$\textbf{\textit{Subject to}}\quad \sum\nolimits_{i\epsilon N} x_{ij} = 1 \quad \forall j\epsilon N \tag{4}$$

$$\sum\nolimits_{j\epsilon N} x_{ij} = 1 \quad \forall i\epsilon N \tag{5}$$

$$a_i - a_j + Nx_{ij} \le N - 1 \quad \forall i, j |\, (i,j)\,\epsilon A - \{1\}\,, \; i \ne j \tag{6}$$

$$0 \le p^f_i \le v^f_i \quad \forall i\epsilon N \tag{7}$$

$$0 \le p^l_i \le \max(0, v^l_i - D_i) \quad \forall i\epsilon N \tag{8}$$

$$0 \le d^l_i \le \max(0, D_i - v^l_i) \quad \forall i\epsilon N \tag{9}$$

$$0 \le p^u_i \le \max(0, v^u_i - \underline{C}_i) \quad \forall i\epsilon N \tag{10}$$

$$h^f_j - h^f_i - p^f_j + M\left(1 - x_{ij}\right) \ge 0 \quad \forall i, j |\, (i,j)\,\epsilon A, j \ge 2 \tag{11}$$

$$h_i^f - h_j^f + p_j^f + M\left(1 - x_{ij}\right) \geq 0 \quad \forall i,j|(i,j)\epsilon A \tag{12}$$

$$R_i^f = v_i^f - p_i^f \quad \forall i\epsilon N \tag{13}$$

$$h_j^l - h_i^l - p_j^l + d_j^l + M\left(1 - x_{ij}\right) \geq 0 \quad \forall i,j|(i,j)\epsilon A, j \geq 2 \tag{14}$$

$$h_i^l - h_j^l + p_j^l - d_j^l + M\left(1 - x_{ij}\right) \geq 0 \quad \forall i,j|(i,j)\epsilon A \tag{15}$$

$$R_i^l = \max\{v_i^l - D_i, 0\} - p_i^l \quad \forall i\epsilon N \tag{16}$$

$$h_j^u - h_i^u - p_j^u + d_j^u + M\left(1 - x_{ij}\right) \geq 0 \quad \forall i,j|(i,j)\epsilon A, j \geq 2 \tag{17}$$

$$h_i^u - h_j^u + p_j^u - d_j^u + M\left(1 - x_{ij}\right) \geq 0 \quad \forall i,j|(i,j)\epsilon A \tag{18}$$

$$\underline{C}_i - v_i^u + p_i^u - d_i^u + \max\left\{g_i^\theta, 0\right\} - U_i^\theta \leq 0 \quad \forall i\epsilon N, \theta\epsilon\Theta \tag{19}$$

$$R_i^f + R_i^l + v_i^u - p_i^u + d_i^u - g_i^\theta - \bar{C}_i - E_i^\theta \leq 0 \quad \forall i\epsilon N, \theta\epsilon\Theta \tag{20}$$

$$h_i^f + h_i^l + h_i^u \leq B \quad \forall i\epsilon N \tag{21}$$

**Algorithm 1** ACO–ILP Algorithm for the Rebalancing Problem

1 **Input:**
2  complete non-directed graph: $G = (N, A)$
3  distance function: $c$
4  rebalancing cost function: $L$
5  set all of the ACO parameters $\{\varphi, \omega, \rho, \#ants, \#iterations\}$
6 initialize all of the pheromone trails $\tau_0$
7 *iteration_result* = []
8 **for** $z \leftarrow 1$ *to* *#iterations* **do**
9  calculate the probability matrix: $P_z = \left[\tau_{z-1}\right]^\varphi (1/c)^\omega$
10  **for** $k \leftarrow 1$ *to* *#ants* **do**
11  $route_{Ant(k)}[0] \leftarrow 0$
12  **for** $i \leftarrow 1$ *to* $N - 1$ **do**
13  list the nodes to be visited: $N_i^k$
14  normalize the probability of the remaining nodes: $P_{z,i}^k$
15  randomly choose the next node according to the probability: *next_node*
16  $route_{Ant(k)}[i] \leftarrow next\_node$
17  evaluate the cost function of each ant: $L_k$
18  *iteration_result*.add ([min $\{L_k\}$ ; argmin $\{L_k\}$])
19  update the pheromone trails: $\tau_z = (1 - \rho)\,\tau_{z-1} + \Delta\tau_z$

The objective of the short-term rebalancing defined by equation (3) is to minimize the cost of the driving distance and the associated penalty costs of pickup activities, remaining faulty and low-battery e-scooters in the system, unmet demand, and excess e-scooters. Constraints (4)–(6) are routing constraints related to the arrivals and departures at all of the nodes and the elimination of subtours. Equations (7)–(10) are pickup and drop-off constraints. Constraints (11) and (12) represent conservation rules for loading faulty e-scooters onto the rebalancing vehicle, and Constraint (13) requires that the number of remaining faulty e-scooters equals the initial number of faulty e-scooters minus those that have been picked up. Similarly, Constraints (14) and (15) are conservation rules for loading (unloading) low-battery e-scooters onto (from) the rebalancing vehicle, and the remainder is determined via equation (16). The conservation of loading and unloading for usable e-scooters is imposed by Constraints (17) and (18). The unmet demand is calculated via equation (19), which requires the number of usable e-scooters in each cluster to be greater than the threshold ($\underline{C}_i$) after both the rebalancing process and the end of the planning time. The number of excess e-scooters is calculated using equation (20), which prevents one spot from being flooded with e-scooters and thus requires a rapid response. Finally, equation (21) is the rebalancing vehicle capacity constraint.

### F. REBALANCING FORMULATION FOR THE HYBRID ACO–ILP ALGORITHM

The ACO algorithm was first introduced in the early 1990s and is inspired by the pheromone trails used by ants as a form of indirect communication to find the shortest path between food and their nest [84]. Thus, the ACO algorithm mimics this foraging behavior by using artificial ants to iteratively adjust the path according to the transition probability, which is based on the "pheromone" concentration and a visibility function. As mentioned earlier, in this study, ACO was employed to generate feasible route sequences, while ILP solver was employed to optimize the number of pickups and drop-offs for these suggested route sequences individually. In other words, the driving cost (the first term of equation (3)) can be calculated for each feasible route sequence suggested by ACO. However, the remaining terms of equation (3) or the new objective function in equation (27) are optimized by the ILP solver under the constraints (28) – (39). This means that the cost function ($L$) in Algorithm 1 is the sum of the driving cost and the optimal rebalancing cost (equation (27)) computed by the ILP solver for each specific route sequence.

Algorithm 1 shows the procedure of rebalancing optimization via ACO–ILP algorithm. Lines 1–4 describe the inputs for ACO–ILP: a graph with a set of nodes $N$ and set of links $A$, a distance function $c$ that represents the driving distance (taken from Bing Maps (https://www.bing.com/maps)), a rebalancing cost function (comprising the driving distance cost and penalty costs optimized by ILP solver), and the parameters of ACO. In each iteration, several steps are performed: calculation of the transition probability, construction of the route sequence for each ant based on the probability, evaluation of the performance of each ant, storage of the best solution found, and updating of the pheromone

trails that are initialized as having equal weights (values of 1). The detailed formulations in the ACO algorithm are as follows.

$$P_{ij}^{k}(z) = \frac{\left[\tau_{ij}(z)\right]^{\varphi}\left(\eta_{ij}\right)^{\omega}}{\sum_{l\epsilon N_i^k}\left[\tau_{il}(z)\right]^{\varphi}\left(\eta_{il}\right)^{\omega}} \quad \forall j\epsilon N_i^k \tag{22}$$

$$\eta_{ij} = 1/c_{ij} \tag{23}$$

$$\tau_{ij}(z+1) = (1-\rho)\,\tau_{ij}(z) + \Delta\tau_{ij}(z) \tag{24}$$

$$\Delta\tau_{ij}(z) = \sum_{k=1}^{m}\Delta\tau_{ij}^{k}(z) \tag{25}$$

$$\Delta\tau_{ij}^{k}(z) = \begin{cases} \dfrac{1}{L_k}, & \text{The k}^{th} \text{ ant passes between } i \text{ and } j \\ 0, & \text{Otherwise} \end{cases} \tag{26}$$

Here, $P_{ij}^{k}(z)$ is the probability of ant $k$ traveling from the current node $i$ to the next node $j$ in iteration $z$; $N_i^k$ is the set of nodes that ant $k$ has not yet visited; $\eta_{ij}$ is the visibility of node $j$ from node $i$, which is defined as the inverse distance between the two nodes; $\varphi$ and $\omega$ are the importance factors of the pheromone and visibility, respectively. The pheromone is updated using equation (24). $\rho$ and $\Delta\tau_{ij}(z)$ represent the pheromone evaporation coefficient and the total amount of pheromone deposited by all of the ants (i.e., $m$ ants), respectively; and $L_k$ is the cost function (which describes the driving distance cost and penalty costs) for ant $k$. In this study, the ACO algorithm was trained using a Python library, Scikit-opt [85]. $\varphi$, $\omega$, and $\rho$ were set to default values: 1, 2, and 0.1, respectively.

In this section, the objective function is the same as in the previous section (equation (3)), except that the ACO algorithm and ILP solver were utilized to solve the rebalancing vehicle routing problem and pickup/drop-off problem, respectively. In this hybrid case, the ILP solver optimizes the pickups and drop-offs for a known route sequence $(\ldots \rightarrow i \rightarrow j \rightarrow \ldots, \forall i, j\epsilon N)$ given by the ACO, and the ACO then combines these penalty costs with the driving distance cost to improve the route sequence iteratively. Therefore, the ILP formulation of rebalancing for a known route sequence is

$$\begin{aligned} \textbf{\textit{Minimize}}\quad & \beta_1\sum_{j\epsilon N}\left(p_j^f + p_j^l + p_j^u\right) + \beta_2\sum_{j\epsilon N}R_j^f \\ & + \beta_3\sum_{j\epsilon N}R_j^l + \frac{\beta_4}{\Theta}\sum_{j\epsilon N;\theta\epsilon\Theta}U_j^{\theta} \\ & + \frac{\beta_5}{\Theta}\sum_{j\epsilon N;\theta\epsilon\Theta}E_j^{\theta} \end{aligned} \tag{27}$$

$$\textbf{\textit{Subject to}}\quad 0 \le p_j^f \le v_j^f \quad \forall j\epsilon N \tag{28}$$

$$0 \le p_j^l \le \max(0, v_j^l - D_j) \quad \forall j\epsilon N \tag{29}$$

$$0 \le d_j^l \le \max(0, D_j - v_j^l) \quad \forall j\epsilon N \tag{30}$$

$$0 \le p_j^u \le \max(0, v_j^u - \underline{C}_j) \quad \forall j\epsilon N \tag{31}$$

$$h_j^f = h_i^f + p_j^f \quad \forall i, j\epsilon N \tag{32}$$

$$R_j^f = v_j^f - p_j^f \quad \forall j\epsilon N \tag{33}$$

$$h_j^l = h_i^l + p_j^l - d_j^l \quad \forall i, j\epsilon N \tag{34}$$

$$R_j^l = \max(0, v_j^l - D_j) - p_j^l \quad \forall j\epsilon N \tag{35}$$

$$h_j^u = h_i^u + p_j^u - d_j^u \quad \forall i, j\epsilon N \tag{36}$$

$$\underline{C}_j - v_j^u + p_j^u - d_j^u + \max\left\{g_j^{\theta}, 0\right\} - U_j^{\theta} \le 0 \quad \forall j\epsilon N, \theta\epsilon\Theta \tag{37}$$

$$R_j^f + R_j^l + v_j^u - p_j^u + d_j^u - g_j^{\theta} - \bar{C}_j - E_j^{\theta} \le 0 \quad \forall i\epsilon N, \theta\epsilon\Theta \tag{38}$$

$$h_j^f + h_j^l + h_j^u \le B \quad \forall j\epsilon N \tag{39}$$

Here, Constraints (28)–(31) are pickup and drop-off constraints at the current node $j$. Equations (32), (34), and (36) give the accumulated numbers of faulty, low-battery, and usable e-scooters, respectively, that are on the rebalancing vehicle at the current node $j$, defined as the sum of those that are on the rebalancing vehicle at the previous node $i$ and the number picked up or dropped off at the current node $j$. Equations (33) and (35) give the remaining numbers of faulty and low-battery e-scooters, respectively. Equations (37) and (38) give the unmet demand and excess e-scooters, respectively. Finally, equation (39) enforces the rebalancing vehicle capacity constraint. The rebalancing vehicle must start from the depot, and all of the decision variables for this node were thus set to zero, except for the variables relating to picking up usable e-scooters $p_1^u$ and the number of usable e-scooters on the rebalancing vehicle $h_1^u$.

## IV. APPLICATION OF DEMAND AND VARIANCE PREDICTION

### A. DATA COLLECTION AND DESCRIPTION

True demand data is highly desirable for operational planning; however, it is often inaccessible unless operators grant access to extract it from user app activities [30]. Due to data limitations, historical ridership data are commonly used to evaluate the effectiveness of proposed models or frameworks [5], [28], [29], [33], [34], [70]. Similarly, this study employs historical data as a case study, while the potential demand is managed through the safety stock parameter or minimum number of usable e-scooters ($\underline{C}_i$). In practice, the proposed framework in this study, particularly the demand prediction model, may require training using true demand data that includes unmet demand, as discussed by Ham et al. [30].

A case study was performed using an open dataset for shared e-scooters operated in Minneapolis (MN) (https://opendata.minneapolismn.gov). The data cover 961,040 trips taken from May 13 to November 25, 2019. The information on each trip comprises the trip ID, trip duration, trip distance, start/end center line ID, and start/end date time. The name of the street where each trip started and ended was recorded, so the center of this street was used as the trip coordinates. In data cleaning, the exclusion criterion was having data missing, and the inclusion criteria were a trip distance (20 m–10 km), a trip duration (20 s–2 h), within the study period (May 14 to November 24, 2019), and

within the study boundary. The 813,970 trips remaining after data cleaning had an average duration of 13 minutes and an average distance of 1.72 km. As mentioned in the previous section, the trips were clustered using the $k$-means clustering algorithm for 15, 30, and 60 clusters; see Fig. 3. For hourly trip-gap prediction ($\Delta t = 1$ h), there were a total of 4,680 samples: 80% were used for model training and 20% were used for model testing (see Fig. 4). Overfitting was avoided by using random sampling to separate the training data into portions for model construction (75%) and model evaluation (25%).

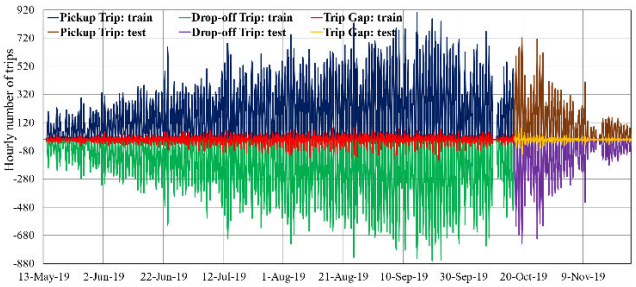


**FIGURE 4.** Hourly pickup and drop-off trips and the trip gap for shared e-scooters in Minneapolis, MN.

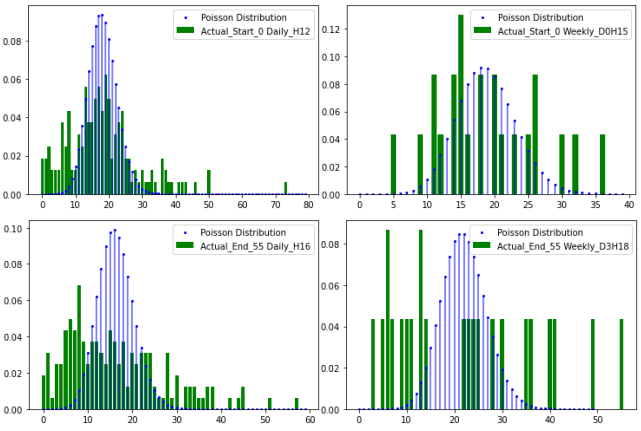


**FIGURE 5.** Histograms and Poisson distributions of the pickup and drop-off demands of shared e-scooters.

Studies have shown that the demand for shared e-scooters correlates with weather attributes, public holidays, annual festivals, and weekly patterns. Therefore, seven hourly weather attributes (temperature, precipitation, wind speed, humidity, wind gust, pressure, and dew point) were downloaded from Weather Underground (www.wunderground.com). Linear interpolated values were to replace values missing from these attributes. In addition, the trip gap prediction model considered official public holidays, annual festivals and events (open street events, a pride festival parade, a state fair festival, a stone arch bridge festival, and an uptown art fair), and the fact that the use of shared e-scooters was banned on October 10, 2019, during the state visit of the President of the United States to Minneapolis. The demand of shared e-scooters is affected by many factors. Thus, it is not suitable to assume that pickup and drop-off demands follow a specific distribution (e.g., a Poisson distribution), see Fig. 5. This characteristic was confirmed by a goodness-of-fit test of daily (same hour of the day) and weekly (same hour of the day and day of the week) patterns. Therefore, the rebalancing approaches based on this assumption (e.g., queue theory) that have commonly been applied for studying shared bike services may not be appropriate for studying shared e-scooter services.

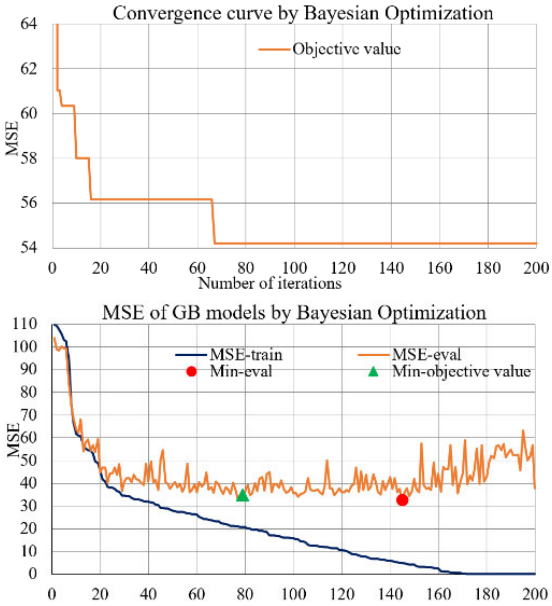


**FIGURE 6.** Hyperparameter optimization by Bayesian Optimization for trip gap prediction.

### B. RESULTS OF DEMAND PREDICTION

As mentioned, the $k$-means algorithm was used to group the trips in clusters of 15, 30, and 60. To improve the performance of the trip gap prediction model, it was trained spatially independently, whereas the model's inputs included external features (as presented in Section IV-A), local historical data, and historical data from four neighboring clusters. Fig. 6 (*top*) shows the convergence curve of the GB hyperparameter tuning performed by BO for cluster 37. Hyperparameter tuning is commonly performed to minimize a loss value on the evaluation dataset (e.g., *MSE_eval*), which likely leads to overfitting, especially for the family of decision tree models. Fig. 6 (*bottom*) plots the BO's evaluated GB models following the sorted value of *MSE_train*. This graph reveals that the minimum *MSE_eval* is located in the region where the deviation between *MSE_train* and *MSE_eval* is high, after which *MSE_eval* worsens or loses generalization. Moreover, the GB modeling in this region is time-consuming, as the models typically require more and deeper decision-tree regressors and a longer lookback length. Therefore, our objective function for BO, which includes the ratio of

**TABLE 2. Results of trip gap prediction and variance prediction.**

| | Model | | 15 Clusters | 30 Clusters | 60 Clusters |
|---|---|---|---|---|---|
| Trip Gap Prediction | HA | RMSE-train | 7.13 | 4.97 | 3.46 |
| | | RMSE-test | 5.25 | 3.94 | 2.66 |
| | DHA | RMSE-train | 6.00 | 4.47 | 3.17 |
| | | RMSE-test | 5.10 | 3.73 | 2.55 |
| | GB | RMSE-train | 5.04 | 3.69 | 2.67 |
| | | RMSE-test | 4.14 | 3.16 | 2.24 |
| Variance Prediction for GB residuals | Constant Variance | Mean-STD | 4.01 | 3.02 | 2.30 |
| | | Coverage | 95.73% | 96.15% | 97.09% |
| | Daily Variance | Mean-STD | 3.44 | 2.57 | 1.93 |
| | | Coverage | 96.13% | 95.65% | 95.96% |
| | SGARCH Variance | Mean-STD | 3.17 | 2.31 | 1.59 |
| | | Coverage | 93.46% | 91.55% | 94.40% |

Remark: HA is historical average, DHA is daily historical average, and GB is gradient boosting.

*MSE_eval* and *MSE_train* has a shorter training time and suggests a more generalized GB model.

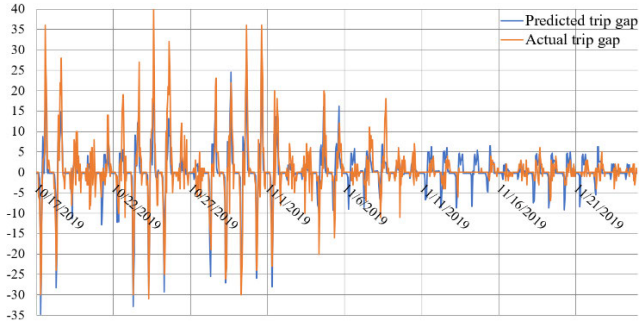


**FIGURE 7. Trip gap predicted using the testing data for cluster 37.**

Fig. 7 presents the trip gap predicted by the GB regressor for cluster 37. The GB model extracted the temporal pattern, but there were prediction errors or residuals. Ignoring such errors in rebalancing planning results in a lower service level or profit than if such errors are taken into account. Table 2 compares the GB prediction with the predictions of two benchmark models (historical average and daily historical average models), with the RMSE used as the accuracy metric. Our numerical results on the training data indicate that GB yielded a smaller RMSE of approximately 26% and 16% compared to the baseline historical and daily historical averages, respectively. With the testing dataset, the GB model's RMSE (or uncertainty) was approximately 19% and 15% lower than the RMSEs of the historical average model and daily historical average model, respectively.

### C. RESULTS OF VARIANCE PREDICTION

The prediction of variance using SGARCH provides two critical parameters for Monte Carlo simulation: the standard deviation (STD) and distribution of the residuals. As explained in Section III-D, a low STD or low uncertainty reduces the expected loss. However, the STD should be minimized such that the coverage (i.e., the percentage of residuals lies within the confidence bounds) is not reduced. For instance, the daily variance of GB residuals had an average STD (Mean-STD) smaller than that of the constant variance without a reduction in coverage (see Table 2). This indicates that the variance prediction model could affect rebalancing planning, as it could further reduce the uncertainty from the demand prediction model. However, the daily variance based on historical data could not capture the long-term fluctuation, as shown in Fig. 8. We also observed that SGARCH variance was more flexible over time (seasonal or annual pattern) than daily variance, as a higher weight was given to the later residuals than the earlier residuals. Overall, compared with the daily variance, the SGARCH model had a slightly smaller Mean-STD but also slightly reduced the coverage.

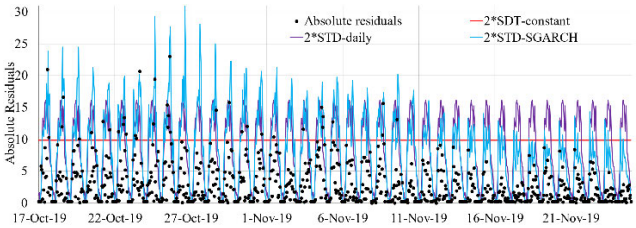


**FIGURE 8. Variance prediction based on residuals of the GB model for cluster 37.**

## V. RESULTS OF REBALANCING OPTIMIZATION

### A. PARAMETER SETTINGS

The open dataset contains trip data that does not cover all the operational planning parameters. Thus, the parameters were simulated using fixed and random values in specified ranges (see Table 3). The coordinates of trips taken by dockless shared e-scooters were partitioned using the $k$-means clustering technique. Increasing the number of clusters dealt with Assumption 1 but reduced the performance of demand prediction models (sparsity and random walk pattern) and exponentially increased the rebalancing optimization time. The number of clusters was set at 15, 30, and 60, and the total number of nodes ($N$) (including the depot and charging stations for these three cases) were 18 (324 edges), 35 (1225 edges), and 70 (4900 edges), respectively. The driving distance from one node to another was obtained from Bing Maps. The computational times for these three cases were 20, 40, and 60 minutes, respectively. The testing data were for a period of approximately 39 days or approximately five weeks, and 30 instances were randomly drawn from the high-demand period (10 am to 8 pm) in the first (for 15-cluster case), middle (for 30-cluster case), and last (for 60-cluster case) three weeks. Monte Carlo simulation was conducted to generate 100 scenarios for our method, whereas the benchmark cases using daily or weekly historical trip data had approximately 180 and 26 scenarios, respectively. One rebalancing vehicle with a capacity of 35 e-scooters was used to complete the rebalancing.

The total number of e-scooters was assumed to be 400, including 20 faulty e-scooters (∼5%) and 60 low-battery

**TABLE 3. Parameter settings for the rebalancing optimization.**

| Parameter | 15-Cluster | 30-Cluster | 60-Cluster |
|---|---|---|---|
| Computational time (minutes) | 20 | 40 | 60 |
| Testing week | 1st, 2nd, 3rd | 2nd, 3rd, 4th | 3rd, 4th, 5th |
| Number of scenarios $\theta$ | 100 | 100 | 100 |
| Number of charging stations | 2 | 4 | 9 |
| Total e-scooters | 400 | 400 | 400 |
| Number of docks in each station (total of 100) | 50 | 25 | 10–15 |
| Maximum number of e-scooters in each cluster $\bar{C}_i$ (total of 700) | 30–50 | 15–30 | 10–15 |
| Minimum number of e-scooters in each cluster $\underline{C}_i$ (total of 80) | 0–10 | 0–5 | 0–3 |
| Number of faulty e-scooters in each cluster $h_i^f$ (total of 20) | 0–3 | 0–2 | 0–2 |
| Number of low-battery e-scooters in each cluster $h_i^l$ (total of 60) | 0–10 | 0–5 | 0–3 |
| Number of usable e-scooters in each cluster $h_i^u$ (total of 320) | 5–25 | 1–20 | 0–10 |
| Vehicle capacity $B$ | 35 | 35 | 35 |
| Unit cost of the driving distance $\beta_0$ | 1 | 1 | 1 |
| Unit cost of picking up e-scooters $\beta_1$ | 0.1 | 0.1 | 0.1 |
| Unit cost of remaining faulty e-scooters $\beta_2$ | 5 | 5 | 5 |
| Unit cost of remaining low-battery e-scooters $\beta_3$ | 3 | 3 | 3 |
| Unit cost of unmet demands $\beta_4$ | 2 | 2 | 2 |
| Unit cost of excess e-scooters $\beta_5$ | 1 | 1 | 1 |

e-scooters (∼15%). Hence, the total number of usable e-scooters was 320, approximately half the hourly pickup demand during the high-demand period. These faulty, low-battery, and usable e-scooters were randomly distributed across the clusters. The average usage time of shared e-scooters in Minneapolis reveals that users spent approximately 3 USD per trip. In the current study, the unit penalty cost of the unmet demand was set as 2 USD, which was approximately 67% of the revenue. The unit cost of excess e-scooters was 1 USD, whereas the penalty costs of the remaining faulty and low-battery e-scooters were 5 and 3 USD, respectively. The driving distance was charged at a rate of 1 USD per kilometer. The pickup cost for each e-scooter was set as 0.1 USD to prevent pickup and drop-off of the same e-scooter from being performed at the same location and to balance the service level. The pickup cost was approximately 5% of the unmet demand penalty, representing a service level Type II (i.e., percentage of served demand) of 95% (i.e., a demand with a probability of less than 5% does not justify the pickup cost).

### B. EXPERIMENTAL ANALYSIS

The rebalancing optimization was performed in the Spyder integrated development environment in Python. The ILP solver GLPK of the Python library Pyomo [86] was used to solve the ILP rebalancing formulation, as in Section III-E. The pickup and drop-off operations in Section III-F were also solved using this ILP solver, whereas the ACO algorithm was trained using another Python library, Scikit-opt [85]. The whole process was performed in a Windows 10 environment on a 64-bit operating system running on an Intel processor core i7-9750H CPU @ 2.60 GHz and equipped with 8.00 GB RAM.

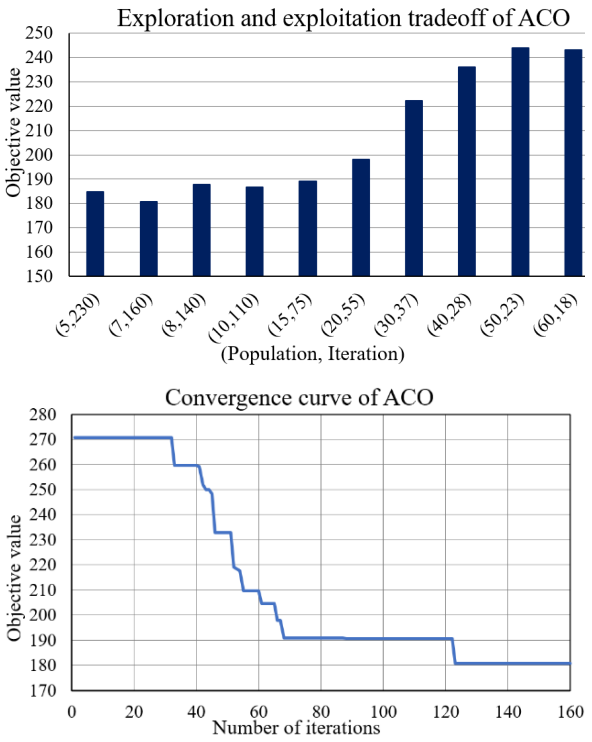


**FIGURE 9. Exploration and exploitation tradeoff of ant colony optimization (top) and the convergence curve (bottom).**

Table 3 shows that the computational time of the balancing optimization was restricted. Therefore, the exploration and exploitation of the hybrid ACO–ILP algorithm were balanced using two ACO parameters, i.e., the population size and number of iterations. A larger population requires fewer iterations than a smaller population to complete an optimization procedure within a limited computational time. Fig. 9 shows the tradeoff between the population size and number of iterations on the top and the convergence curve of the optimal trial (population size = 7 and number of iterations = 160) on the bottom. The tradeoff was examined independently for 10 trials of the 15-cluster, 30-cluster, and 60-cluster problems. The optimal ant populations for these three problems were 65, 90, and 7, and the number of iterations were 25, 20, and 160, respectively. The benchmark cases (actual, historical daily, and historical weekly trip gaps) had different numbers of scenarios, and the population size was increased or reduced accordingly. For 15-cluster problems, the population sizes were set to 250, 50, and 150 for actual, daily, and weekly scenarios, respectively. Similarly, for 30-cluster and 60-cluster problems, the population sizes of these three scenarios were (300, 75, and 200) and (50, 5, and 20), respectively.

Fig. 10 shows the optimal rebalancing result of an instance in the 15-cluster problem. The penalty cost for the remaining

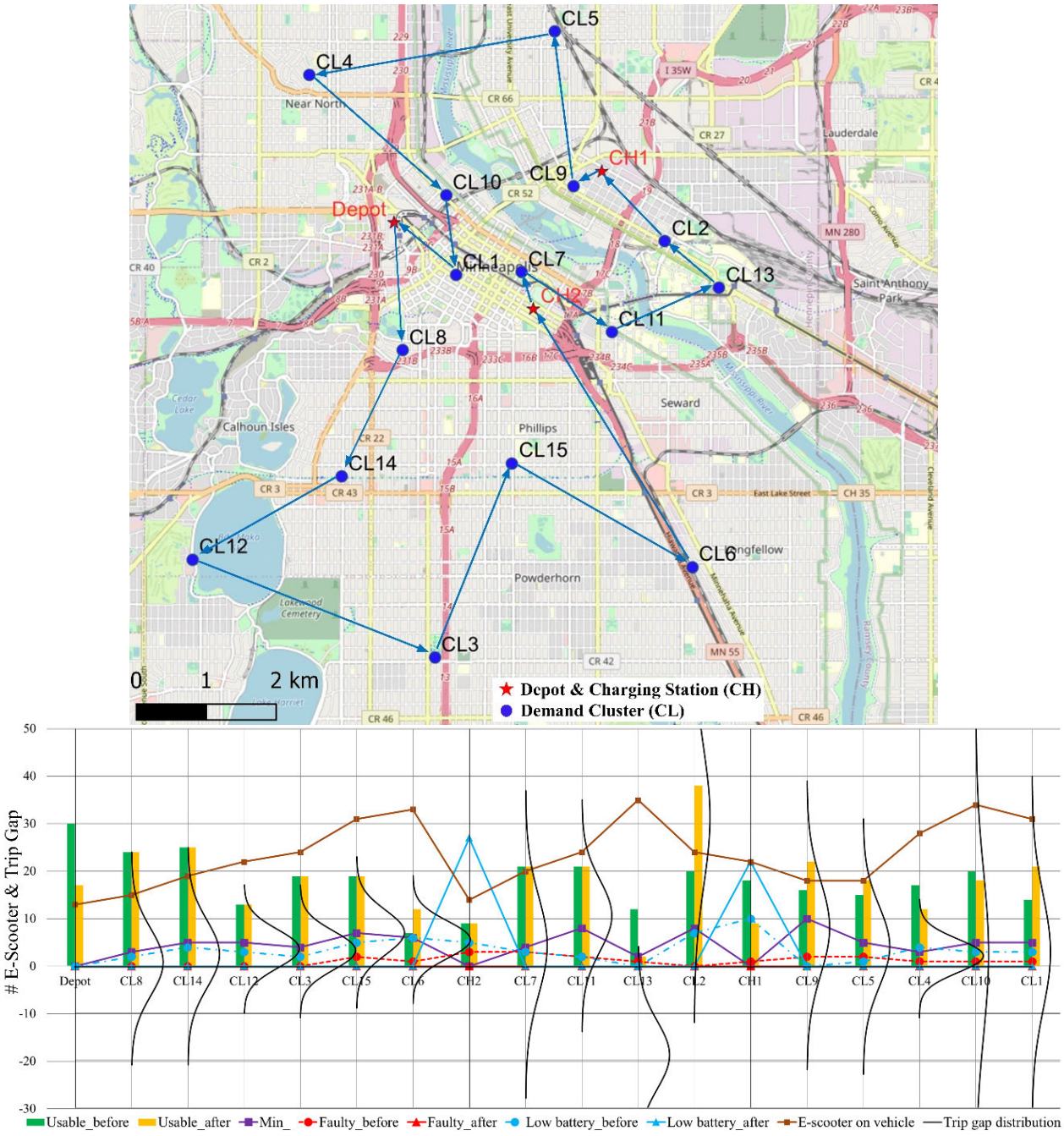


**FIGURE 10.** (Top) Optimal route sequence of an instance in the 15-cluster problem and (bottom) its optimal pickup and drop-off results (CH: charging station, CL: cluster center of aggregated trips by *k*-means algorithm.)

faulty and low-battery e-scooters was high, so there was no remainder after rebalancing. In this case, the vehicle passed through several demand clusters and collected the low-battery e-scooters before dropping them off at the charging stations.

At the first couple of nodes, the rebalancing vehicle took 13 usable e-scooters from the depot, then picked up 22 low-battery e-scooters and delivered them to charging station 2. At cluster 6, five usable e-scooters were dropped off, which might have been unnecessary if we had ignored the demand uncertainty. The remainder (eight usable e-scooters) and the other ten usable e-scooters picked up at cluster 13 were dropped off at cluster 2, which had a considerable trip gap variation. The accumulated low-battery e-scooters at cluster 2 were then dropped off at charging station 1, where nine usable e-scooters were picked up and relocated to either cluster 9 (six e-scooters) or cluster 5 (three e-scooters). Finally, five and two usable e-scooters were relocated from clusters 4 and 10, respectively, to cluster 1, as this possibly gave a higher expected revenue. The accumulated faulty and low-battery e-scooters on the vehicle up to this node were

**TABLE 4. Results of rebalancing optimization for each problem size.**

| Problem Size | Optimization Approach | Case | Computational Time (minute) | Objective Value (USD) | Driving Distance (km) | Service Level | Remaining Faulty E-scooters | Remaining Low-Battery E-scooters |
|---|---|---|---|---|---|---|---|---|
| 15-Cluster | ILP solver | Actual | 20.00 | 65.00 | 50.41 | 99.33% | 0.00% | 0.00% |
| | | Sampling | 20.00 | 66.66 | 49.71 | 96.22% | 0.00% | 0.00% |
| | | Daily | 20.00 | 71.18 | 50.03 | 97.33% | 0.00% | 0.00% |
| | | Weekly | 20.00 | 68.85 | 50.03 | 97.33% | 0.00% | 0.00% |
| | ACO–ILP | Actual | 11.44 | 65.43 | 50.66 | 99.33% | 0.00% | 0.00% |
| | | Sampling | 20.67 | 71.26 | 53.94 | 96.44% | 0.00% | 0.00% |
| | | Daily | 27.04 | 78.17 | 56.84 | 97.11% | 0.00% | 0.06% |
| | | Weekly | 15.06 | 70.17 | 51.11 | 96.89% | 0.00% | 0.00% |
| 30-Cluster | ILP solver | Actual | 40.00 | 92.73 | 80.09 | 99.89% | 0.00% | 0.06% |
| | | Sampling | 40.00 | 124.40 | 91.64 | 96.44% | 0.00% | 0.00% |
| | | Daily | 40.00 | 135.58 | 93.44 | 96.44% | 0.00% | 0.00% |
| | | Weekly | 40.00 | 127.86 | 89.98 | 96.44% | 0.00% | 0.00% |
| | ACO–ILP | Actual | 9.53 | 100.08 | 86.82 | 99.67% | 0.00% | 0.22% |
| | | Sampling | 40.29 | 129.42 | 97.64 | 96.67% | 0.00% | 0.11% |
| | | Daily | 53.76 | 144.87 | 102.86 | 96.67% | 0.00% | 0.06% |
| | | Weekly | 19.78 | 130.49 | 93.03 | 96.67% | 0.00% | 0.11% |
| 60-Cluster | ILP solver | Actual | 60.00 | 147.97 | 136.16 | 100.00% | 0.00% | 0.00% |
| | | Sampling | 60.00 | 199.82 | 159.24 | 97.39% | 0.00% | 0.11% |
| | | Daily | 119.58 | 257.60 | 179.63 | 97.06% | 1.00% | 1.06% |
| | | Weekly | 72.00 | 245.44 | 175.76 | 96.94% | 0.00% | 0.89% |
| | ACO–ILP | Actual | 20.25 | 135.74 | 123.88 | 99.94% | 0.00% | 0.00% |
| | | Sampling | 59.99 | 167.33 | 127.57 | 97.72% | 0.00% | 0.00% |
| | | Daily | 82.39 | 200.81 | 127.27 | 97.28% | 0.00% | 0.00% |
| | | Weekly | 36.34 | 192.94 | 126.24 | 97.33% | 0.00% | 0.00% |

then taken back to the depot for repair and recharging. This result shows that the expected unmet demand was minimized under several constraints (e.g., the vehicle capacity, driving distance, and the available number of usable e-scooters) by relocating the usable e-scooters from a location with an excessive number of e-scooters or a low expected demand to a location with a higher expected demand.

The ILP solver provided a feasible solution for all of the instances of 15- and 30-cluster problems but did not provide a feasible solution for the 60-cluster problem in some instances of historical daily (70%) and historical weekly (25%) trip gaps. There were approximately 200 and 28 scenarios for historical daily and historical weekly trip gaps, respectively, whereas the Monte Carlo sampling had 100 scenarios. Our results from Monte Carlo sampling demonstrate that the ILP solver can generate feasible solutions within the given computational time constraints for this large-scale problem, even with up to 500 simulated scenarios. Hence, the distribution and variance of the trip gaps rather than just the number of scenarios contributed to the complexity of the optimization problem, which prevented the ILP solver from reaching a feasible solution. To obtain feasible results for all instances of historical daily and historical weekly trip gaps, the computational time was iteratively increased by 30 and 15 minutes, respectively. Table 4 shows that the computational time required to reach a feasible solution in these two cases was 120 and 72 minutes, respectively.

In the case of ACO–ILP, the computational time for actual cases with only one demand scenario is significantly reduced despite increasing the population sizes by more than three times compared to those of the sampling method. Similarly, the computational time for historical weekly demand was slightly shorter than the allotted time, even with a population size increase of approximately 2–3 times. In contrast, the computational time for the historical daily baseline was slightly longer than the target training time.

Table 4 presents the average objective values for all three cluster problems based on 30 random instances. This includes the overall objective value, driving distance, service level, and the count of remaining faulty and low-battery e-scooters. This table shows that the ILP solver outperformed for small-sized problems (15- and 30-cluster problems), whereas the hybrid ACO–ILP algorithm achieved better objective values for large-sized problems (60-cluster problems). The table also indicates that the average service level (Type II) is 96.91%, slightly surpassing the expected service level of 95%, as presented in the previous section. However, the service level could substantially improve with perfect information, i.e., actual trip gaps. To achieve this target service level, the driving distance is proportionally increased based on the number of nodes or the size of the rebalancing problems. For instance, the driving distances are approximately 50 km, 92 km, and 127 km for 15-, 30-, and 60-cluster problems, respectively.

From Table 4, it is observed that the occurrences of remaining faulty and low-battery e-scooters were more prominent with the less optimal optimization algorithm, i.e., ACO–ILP for the 15- and 30-cluster problems and ILP solver

for the 60-cluster problems. Given the relatively high penalty cost associated with these two types of e-scooters, there were minimal remaining faulty and low-battery e-scooters after the rebalancing operation. Notably, the penalty cost of remaining faulty e-scooters is higher than that of low-battery e-scooters, resulting in a higher percentage of remaining low-battery e-scooters than faulty ones. In other words, imposing a higher penalty cost on these e-scooters results in a lower rate of remaining faulty and low-battery e-scooters, potentially leading to longer driving distances. For practical implications, conducting a sensitivity analysis on these two parameters, e.g., focusing on the tradeoff between driving distance and the acceptable percentage of remaining faulty and low-battery e-scooters, is imperative.

Assumption 3 requires a high number of clusters, but this leads to a long driving distance for the rebalancing vehicle and high penalty costs due to the high demand uncertainty. This increased uncertainty can also be observed in Table 2, as the reduction in RMSE is not proportional to the number of clusters. The Elbow method is commonly employed to select an appropriate number of clusters, which can be further increased until the radius of the clusters falls within the walkable range of 200 to 500 meters. However, using the perfect information (actual trip gaps) had the lowest penalty cost irrespective of the number of clusters. According to the average optimal objective value of the 15-cluster problems obtained using the ILP solver, the sampling approach had an objective value 6.78% lower than that when using the historical daily trip gaps and 3.29% lower than that when using the historical weekly trip gaps. In the case of the 30-cluster problems, the proposed framework had an objective value of 8.99% and 2.78% smaller than the objective values of these two benchmarks, respectively. In the case of the 60-cluster problems, the objective value obtained using the hybrid algorithm was 20.01% less than that when using the historical daily trip gaps and 15.31% less than that when using the historical weekly trip gaps. In summary, across all three cases, the proposed framework reduced the objective value by 11.93% and 7.12% compared to the baseline historical daily and weekly cases, respectively. However, the perfect information, i.e., actual trip gaps, demonstrated a more substantial reduction of approximately 15.61%. Therefore, the demand uncertainty of shared e-scooters could be further minimized either by employing comprehensive deep-learning models or by incorporating additional explanatory features.

### C. DISCUSSION

During the pilot program in 2019, the city of Minneapolis allowed up to 2,000 shared e-scooters, which were deployed by several operators, such as Lime, Lyft, JUMP, and Spin. The present study set the total number of e-scooters as 400, which was approximately the number of e-scooters deployed by each operator. In the case of dockless shared bikes, Hua et al. [87] found that 96.8% of the trip demand could be satisfied by supplying just 14.5% of the original fleet. Under a limitation of 400 e-scooters (approximately 20% of the allowed fleet), we achieved a service level of approximately 96.91% with hourly rebalancing. From an operational viewpoint, this means that we could reduce the deployed fleet if all of the operators participated in unified rebalancing planning or reduced the overlap of their deployment areas by using geofences. From an emissions viewpoint, frequent rebalancing, especially when a vehicle with an internal combustion engine visits many clusters, is a poor option because servicing can account for 50% of total emissions [19]. It is also impractical to unify operations among operators, so properly separating deployment zones (that minimize overlap and are partitioned by maximizing intrazonal trips) is a promising choice. In this case, the rebalancing should be performed only a few times per day, and the operators should use an electric rebalancing vehicle and minimize the number of clusters to visit.

The presented empirical findings demonstrate that optimal driving distances approximate 50 km, 92 km, and 127 km for the three distinct problem sizes: 15-cluster, 30-cluster, and 60-cluster cases. As a result, the case study indicates that a single rebalancing vehicle is unable to complete the rebalancing procedure within the allotted one-hour time-frame. Under these circumstances, our proposed framework exhibits greater feasibility for extended rebalancing periods, such as 2 to 3 hours, or it might necessitate certain adjustments to fulfill specific objectives, such as driving distance reduction or service level enhancement. The first modification involves allowing the rebalancing vehicle to bypass unnecessary nodes, e.g., nodes with low penalty costs. The second modification suggests increasing the number of rebalancing vehicles, especially using multiple small ones. However, the most advisable course of action is the third option. This involves the elimination of balanced demand clusters before initiating the rebalancing optimization and rebalancing operation. By doing so, the computational time and the driving distance can be reduced as the number of nodes becomes smaller. In this case, the operators can focus only on nodes with a significant deviation from the desired supply level, i.e., the sum of predicted trip gaps, predicted trip gap variations with service level parameters, and safety stock. When the number of nodes is relatively small, the rebalancing optimization can be solved by an ILP solver; otherwise, employ ACO–ILP with parallel computing.

## VI. CONCLUSION AND FUTURE STUDIES

In summary, dockless shared e-scooters are a potential solution for compact urban mobility, particularly for addressing the first-mile–last-mile problem, parking shortages, and providing an alternative mode of transportation. However, the trip characteristics, vehicle characteristics, and regulations make the short-term operational planning of this transportation mode more challenging than that of the most similar transportation mode, namely shared bikes. For instance, the pickup and drop-off demands of shared e-scooters do not follow typical Poisson distributions, which means that the

operational planning approaches commonly applied under demand uncertainty to shared bikes (e.g., the use of Markov chain or queue theory) may not be appropriate for shared e-scooters. Therefore, this study proposed a new framework for the short-term rebalancing of this mode, which involved Monte Carlo simulation based on the predicted demand and variance. The GB model was used to forecast the hourly trip gap of shared e-scooters, and SGARCH trained the model's residuals. As a result, the integration of two models (GB and SGARCH) reduced the demand uncertainty in terms of the RMSE and average STD. Moreover, a new ILP rebalancing formulation was constructed by considering the demand uncertainty, faulty e-scooters, low-battery e-scooters, and distribution regulations. This is an NP-hard problem, so an ILP solver was used to solve the small-size problem, whereas the proposed hybrid ACO–ILP algorithm was used to solve the large-size problem. The numerical results for the most practical case (the 60-cluster case) based on real-world data for Minneapolis, MN, show that the use of our framework reduced the operational burden by 20.01% and 15.30% relative to the benchmark practices of using the historical daily and weekly demands, respectively. Furthermore, the ILP solver sometimes did not provide a feasible solution under a computational time constraint for these two benchmarks, but this problem did not arise with our simulated demand. In other words, the ILP solver required longer computational time to produce a feasible solution for the baseline problems than our Monte Carlo sampling approach.

Future studies could focus on dynamic rebalancing, including rebalancing planning for a few steps ahead or in real time. The stochasticity of faulty and low-battery e-scooters may also be considered in future research, especially if relevant information (accumulated riding duration, battery drainage rate, etc.) is available. Moreover, for more practical applications, future research could implement parallel computing to reduce the computational time of the rebalancing optimization, specifically for large-scale problems. The proposed remedial suggestions, which include using multiple smaller rebalancing vehicles, allowing the rebalancing vehicle to skip specific nodes selectively, and getting rid of balanced clusters, merit thorough examination in future experimental studies. Since SGARCH is an exponentially weighted average of the prior squared errors, this model is highly susceptible to significant errors from the demand prediction model. Hence, the future research direction could involve implementing machine learning or deep learning algorithms to deal with this limitation or obtain lower demand uncertainty than is currently achievable.

## REFERENCES

[1] N. Saum and M. Piantanakulchai, "A review on an emerging new mode of transport: The shared dockless electric scooter," in *Proc. East. Asia Soc. Transp. Stud.*, Colombo, Sri Lanka, 2019, p. 137. [Online]. Available: https://www.easts.info/on-line/proceedings/vol.12/head.htm

[2] W. Riggs, M. Kawashima, and D. Batstone, "Exploring best practice for municipal e-scooter policy in the United States," *Transp. Res. A, Policy Pract.*, vol. 151, pp. 18–27, Sep. 2021, doi: 10.1016/j.tra.2021.06.025.

[3] T. Benarbia, K. Labadi, A. M. Darcherif, J.-P. Barbot, and A. Omari, "Real-time inventory control and rebalancing in bike-sharing systems by using a stochastic Petri net model," in *Proc. 3rd Int. Conf. Syst. Control*, Algiers, Algeria, 2013, pp. 583–589.

[4] NACTO. (2019). *Shared Micromobility in the US*. [Online]. Available: https://nacto.org/shared-micromobility-2019/

[5] A. Hosseinzadeh, A. Karimpour, and R. Kluger, "Factors influencing shared micromobility services: An analysis of e-scooters and bikeshare," *Transp. Res. D, Transp. Environ.*, vol. 100, Nov. 2021, Art. no. 103047, doi: 10.1016/j.trd.2021.103047.

[6] K. Wang, X. Qian, D. T. Fitch, Y. Lee, J. Malik, and G. Circella, "What travel modes do shared e-scooters displace? A review of recent research findings," *Transp. Rev.*, vol. 43, no. 1, pp. 5–31, Jan. 2023, doi: 10.1080/01441647.2021.2015639.

[7] A.-H. Kirstin, B. Brandon, O. N. Riley, and S. Smith, "Governing micro-mobility: A nationwide assessment of electric scooter regulations," in *Proc. Transp. Res. Board 98th Ann. Meeting*, Washington DC, USA, Jan. 2019, Paper 19-05267.

[8] K. Button, H. Frye, and D. Reaves, "Economic regulation and e-scooter networks in the USA," *Res. Transp. Econ.*, vol. 84, Dec. 2020, Art. no. 100973, doi: 10.1016/j.retrec.2020.100973.

[9] A. Brown, "Micromobility, macro goals: Aligning scooter parking policy with broader city objectives," *Transp. Res. Interdiscipl. Perspect.*, vol. 12, Dec. 2021, Art. no. 100508, doi: 10.1016/j.trip.2021.100508.

[10] A. Pashkevich, T. E. Burghardt, S. Pulawska-Obiedowska, and M. Sucha, "Visual attention and speeds of pedestrians, cyclists, and electric scooter riders when using shared road—A field eye tracker experiment," *Case Stud. Transp. Policy*, vol. 10, no. 1, pp. 549–558, Mar. 2022, doi: 10.1016/j.cstp.2022.01.015.

[11] S. Sareen, D. Remme, and H. Haarstad, "E-scooter regulation: The micro-politics of market-making for micro-mobility in Bergen," *Environ. Innov. Societal Transitions*, vol. 40, pp. 461–473, Sep. 2021, doi: 10.1016/j.eist.2021.10.009.

[12] A. Brown, N. J. Klein, C. Thigpen, and N. Williams, "Impeding access: The frequency and characteristics of improper scooter, bike, and car parking," *Transp. Res. Interdiscipl. Perspect.*, vol. 4, Mar. 2020, Art. no. 100099, doi: 10.1016/j.trip.2020.100099.

[13] H. Li, Z. Yuan, T. Novack, W. Huang, and A. Zipf, "Understanding spatiotemporal trip purposes of urban micro-mobility from the lens of dockless e-scooter sharing," *Comput., Environ. Urban Syst.*, vol. 96, Sep. 2022, Art. no. 101848, doi: 10.1016/j.compenvurbsys.2022.101848.

[14] C. S. Smith and J. P. Schwieterman, "E-scooter scenarios: Evaluating the potential mobility benefits of shared dockless scooters in Chicago," Chaddick Inst. Metrop. Develop. DePaul Univ., Chicago, IL, USA, Tech. Rep., 2018. [Online]. Available: https://www.itskrs.its.dot.gov/its/benecost.nsf/ID/a0081cb53992c2d6852585690067c6e4

[15] O. Caspi, M. J. Smart, and R. B. Noland, "Spatial associations of dockless shared e-scooter usage," *Transp. Res. D, Transp. Environ.*, vol. 86, Sep. 2020, Art. no. 102396, doi: 10.1016/j.trd.2020.102396.

[16] R. Zhu, X. Zhang, D. Kondor, P. Santi, and C. Ratti, "Understanding spatio-temporal heterogeneity of bike-sharing and scooter-sharing mobility," *Comput., Environ. Urban Syst.*, vol. 81, May 2020, Art. no. 101483, doi: 10.1016/j.compenvurbsys.2020.101483.

[17] H. Younes, Z. Zou, J. Wu, and G. Baiocchi, "Comparing the temporal determinants of dockless scooter-share and station-based bike-share in Washington, DC," *Transp. Res. A, Policy Pract.*, vol. 134, pp. 308–320, Apr. 2020, doi: 10.1016/j.tra.2020.02.021.

[18] G. McKenzie, "Urban mobility in the sharing economy: A spatiotemporal comparison of shared mobility services," *Comput., Environ. Urban Syst.*, vol. 79, Jan. 2020, Art. no. 101418, doi: 10.1016/j.compenvurbsys.2019.101418.

[19] A. de Bortoli and Z. Christoforou, "Consequential LCA for territorial and multimodal transportation policies: Method and application to the free-floating e-scooter disruption in Paris," *J. Cleaner Prod.*, vol. 273, Nov. 2020, Art. no. 122898, doi: 10.1016/j.jclepro.2020.122898.

[20] H. Moreau, L. de Jamblinne de Meux, V. Zeller, P. D'Ans, C. Ruwet, and W. M. J. Achten, "Dockless e-scooter: A green solution for mobility? Comparative case study between dockless e-scooters, displaced transport, and personal e-scooters," *Sustainability*, vol. 12, no. 5, p. 1803, Feb. 2020, doi: 10.3390/su12051803.

[21] H. Peng, Y. Nishiyama, and K. Sezaki, "Assessing environmental benefits from shared micromobility systems using machine learning algorithms and Monte Carlo simulation," *Sustain. Cities Soc.*, vol. 87, Dec. 2022, Art. no. 104207, doi: 10.1016/j.scs.2022.104207.

[22] S. Severengiz, S. Finke, N. Schelte, and N. Wendt, "Life cycle assessment on the mobility service e-scooter sharing," in *Proc. IEEE Eur. Technol. Eng. Manag. Summit (E-TEMS)*, Mar. 2020, pp. 1–6.

[23] M. Javadinasr, S. Asgharpour, E. Rahimi, P. Choobchian, A. K. Mohammadian, and J. Auld, "Eliciting attitudinal factors affecting the continuance use of e-scooters: An empirical study in Chicago," *Transp. Res. F, Traffic Psychol. Behaviour*, vol. 87, pp. 87–101, May 2022, doi: 10.1016/j.trf.2022.03.019.

[24] R. G. Öztas Karli, H. Karli, and H. S. Çelikyay, "Investigating the acceptance of shared e-scooters: Empirical evidence from Turkey," *Case Stud. Transp. Policy*, vol. 10, no. 2, pp. 1058–1068, Jun. 2022, doi: 10.1016/j.cstp.2022.03.018.

[25] M. Abouelela, C. Al Haddad, and C. Antoniou, "Are young users willing to shift from carsharing to scooter–sharing?" *Transp. Res. D, Transp. Environ.*, vol. 95, Jun. 2021, Art. no. 102821, doi: 10.1016/j.trd.2021.102821.

[26] H. Fitt and A. Curl, "The early days of shared micromobility: A social practices approach," *J. Transp. Geography*, vol. 86, Jun. 2020, Art. no. 102779, doi: 10.1016/j.jtrangeo.2020.102779.

[27] S. He and K. G. Shin, "Dynamic flow distribution prediction for urban dockless e-scooter sharing reconfiguration," in *Proc. Web Conf.*, Apr. 2020, pp. 133–143.

[28] S. He and K. G. Shin, "Distribution prediction for reconfiguring urban dockless e-scooter sharing systems," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 12, pp. 5722–5740, Dec. 2022, doi: 10.1109/TKDE.2021.3062074.

[29] N. Saum, S. Sugiura, and M. Piantanakulchai, "Short-term demand and volatility prediction of shared micro-mobility: A case study of e-scooter in Thammasat University," in *Proc. Forum Integr. Sustain. Transp. Syst. (FISTS)*, Delft, The Netherlands, Nov. 2020, pp. 27–32, doi: 10.1109/FISTS46898.2020.9264852.

[30] S. W. Ham, J.-H. Cho, S. Park, and D.-K. Kim, "Spatiotemporal demand prediction model for e-scooter sharing services with latent feature and deep learning," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2675, no. 11, pp. 34–43, Apr. 2021, doi: 10.1177/03611981211003896.

[31] S. Phithakkitnukooon, K. Patanukhom, and M. G. Demissie, "Predicting spatiotemporal demand of dockless e-scooter sharing services with a masked fully convolutional network," *ISPRS Int. J. Geo-Inf.*, vol. 10, no. 11, p. 773, Nov. 2021.

[32] Y. Xu, X. Zhao, X. Zhang, and M. Paliwal, "Real-time forecasting of dockless scooter-sharing demand: A spatio-temporal multi-graph transformer approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 8, pp. 8507–8518, Jan. 2023, doi: 10.1109/TITS.2023.3239309.

[33] S. Kim, S. Choo, G. Lee, and S. Kim, "Predicting demand for shared e-scooter using community structure and deep learning method," *Sustainability*, vol. 14, no. 5, p. 2564, Feb. 2022, doi: 10.3390/su14052564.

[34] P. W. Khan, S.-J. Park, S.-J. Lee, and Y.-C. Byun, "Electric kickboard demand prediction in spatiotemporal dimension using clustering-aided bagging regressor," *J. Adv. Transp.*, vol. 2022, pp. 1–15, Aug. 2022, doi: 10.1155/2022/8062932.

[35] M. Masoud, M. Elhenawy, M. H. Almannaa, S. Q. Liu, S. Glaser, and A. Rakotonirainy, "Heuristic approaches to solve e-scooter assignment problem," *IEEE Access*, vol. 7, pp. 175093–175105, 2019, doi: 10.1109/ACCESS.2019.2957303.

[36] A. Ciociola, M. Cocca, D. Giordano, L. Vassio, and M. Mellia, "E-scooter sharing: Leveraging open data for system design," in *Proc. IEEE/ACM 24th Int. Symp. Distrib. Simul. Real Time Appl. (DS-RT)*, Prague, Czech Republic, Sep. 2020, pp. 1–8.

[37] L. Tolomei, S. Fiorini, A. Ciociola, L. Vassio, D. Giordano, and M. Mellia, "Benefits of relocation on e-scooter sharing—A data-informed approach," in *Proc. IEEE Int. Intell. Transp. Syst. Conf. (ITSC)*, Sep. 2021, pp. 3170–3175.

[38] J. Osorio, C. Lei, and Y. Ouyang, "Optimal rebalancing and on-board charging of shared electric scooters," *Transp. Res. B, Methodol.*, vol. 147, pp. 197–219, May 2021, doi: 10.1016/j.trb.2021.03.009.

[39] A. M. Fathabad, X. Li, J. Cheng, and Y.-J. Wu, "Data-driven optimization for e-scooter system design," Nat. Inst. Transp. Communities (NITC), Dept. Syst. Ind. Eng., Univ. Arizona, Tucson, AZ, USA, Tech. Rep., NITC-RR-1382, Jun. 2022. [Online]. Available: https://rosap.ntl.bts.gov/view/dot/62800

[40] G. Losapio, F. Minutoli, V. Mascardi, and A. Ferrando, "Smart balancing of e-scooter sharing systems via deep reinforcement learning," in *Proc. 22nd Workshop, Objects Agents*, Bologna, Italy, 2022, pp. 83–97.

[41] H. Akova, S. Hulagu, and H. B. Celikoglu, "Effects of energy consumption on cost optimal recharging station locations for e-scooters," in *Proc. 7th Int. Conf. Models Technol. Intell. Transp. Syst. (MT-ITS)*, Jun. 2021, pp. 1–6.

[42] O. Altintasi and S. Yalcinkaya, "Siting charging stations and identifying safe and convenient routes for environmentally sustainable e-scooter systems," *Sustain. Cities Soc.*, vol. 84, Sep. 2022, Art. no. 104020, doi: 10.1016/j.scs.2022.104020.

[43] G. McKenzie, "Spatiotemporal comparative analysis of scooter-share and bike-share usage patterns in Washington, DC," *J. Transp. Geography*, vol. 78, pp. 19–28, Jun. 2019, doi: 10.1016/j.jtrangeo.2019.05.007.

[44] A. Martínez-Navarro, V.-A. Cloquell-Ballester, and S. Segui-Chilet, "Photovoltaic electric scooter charger dock for the development of sustainable mobility in urban environments," *IEEE Access*, vol. 8, pp. 169486–169495, 2020, doi: 10.1109/ACCESS.2020.3023881.

[45] C. S. Shui and W. Y. Szeto, "A review of bicycle-sharing service planning problems," *Transp. Res. C, Emerg. Technol.*, vol. 117, Aug. 2020, Art. no. 102648, doi: 10.1016/j.trc.2020.102648.

[46] M. Du, L. Cheng, X. Li, and F. Tang, "Static rebalancing optimization with considering the collection of malfunctioning bikes in free-floating bike sharing system," *Transp. Res. E, Logistics Transp. Rev.*, vol. 141, Sep. 2020, Art. no. 102012, doi: 10.1016/j.tre.2020.102012.

[47] X. Chang, J. Wu, H. Sun, G. H. D. A. Correia, and J. Chen, "Relocating operational and damaged bikes in free-floating systems: A data-driven modeling framework for level of service enhancement," *Transp. Res. A, Policy Pract.*, vol. 153, pp. 235–260, Nov. 2021, doi: 10.1016/j.tra.2021.09.010.

[48] S. Kim, G. Lee, and S. Choo, "Optimal rebalancing strategy for shared e-scooter using genetic algorithm," *J. Adv. Transp.*, vol. 2023, pp. 1–13, Apr. 2023, doi: 10.1155/2023/2696651.

[49] L. Pan, X. Liu, Y. Xia, and L.-N. Xing, "Tabu search algorithm for the bike sharing rebalancing problem," *IEEE Access*, vol. 8, pp. 144543–144556, 2020, doi: 10.1109/ACCESS.2020.3011844.

[50] S. C. Ho and W. Y. Szeto, "A hybrid large neighborhood search for the static multi-vehicle bike-repositioning problem," *Transp. Res. B, Methodol.*, vol. 95, pp. 340–363, Jan. 2017, doi: 10.1016/j.trb.2016.11.003.

[51] A. Pal and Y. Zhang, "Free-floating bike sharing: Solving real-life large-scale static rebalancing problems," *Transp. Res. C, Emerg. Technol.*, vol. 80, pp. 92–116, Jul. 2017, doi: 10.1016/j.trc.2017.03.016.

[52] S. Zhang, G. Xiang, and Z. Huang, "Bike-sharing static rebalancing by considering the collection of bicycles in need of repair," *J. Adv. Transp.*, vol. 2018, pp. 1–18, Sep. 2018, doi: 10.1155/2018/8086378.

[53] Y. Jia, Y. Xu, D. Yang, and J. Li, "The biobjective bike-sharing rebalancing problem with balance intervals: A multistart multiobjective particle swarm optimization algorithm," *Complexity*, vol. 2020, pp. 1–19, Sep. 2020, doi: 10.1155/2020/2845426.

[54] S. Ghosh, P. Varakantham, Y. Adulyasak, and P. Jaillet, "Dynamic repositioning to reduce lost demand in bike sharing systems," *J. Artif. Intell. Res.*, vol. 58, pp. 387–430, Feb. 2017.

[55] M. Dell'Amico, E. Hadjicostantinou, M. Iori, and S. Novellani, "The bike sharing rebalancing problem: Mathematical formulations and benchmark instances," *Omega*, vol. 45, pp. 7–19, Jun. 2014, doi: 10.1016/j.omega.2013.12.001.

[56] T. Raviv and O. Kolka, "Optimal inventory management of a bike-sharing station," *IIE Trans.*, vol. 45, no. 10, pp. 1077–1093, 2013, doi: 10.1080/0740817X.2013.770186.

[57] R. Alvarez-Valdes, J. M. Belenguer, E. Benavent, J. D. Bermudez, F. Muñoz, E. Vercher, and F. Verdejo, "Optimizing the level of service quality of a bike-sharing system," *Omega*, vol. 62, pp. 163–175, Jul. 2016, doi: 10.1016/j.omega.2015.09.007.

[58] E. O'Mahony, "Smarter tools for (Citi) bike sharing," Ph.D. dissertation, Graduate School Cornell Univ., Dept. Comput. Sci., Cornell Univ., New York, NY, USA, 2015.

[59] J. Schuijbroek, R. C. Hampshire, and W.-J. van Hoeve, "Inventory rebalancing and vehicle routing in bike sharing systems," *Eur. J. Oper. Res.*, vol. 257, no. 3, pp. 992–1004, Mar. 2017, doi: 10.1016/j.ejor.2016.08.029.

[60] Y.-H. Seo, "A dynamic rebalancing strategy in public bicycle sharing systems based on real-time dynamic programming and reinforcement learning," Ph.D. dissertation, Dept. Civil Environ. Eng., Seoul Nat. Univ., Seoul, (South) Korea, 2020.

[61] C.-C. Lu, "Robust multi-period fleet allocation models for bike-sharing systems," *Netw. Spatial Econ.*, vol. 16, no. 1, pp. 61–82, Mar. 2016, doi: 10.1007/s11067-013-9203-9.

[62] Y. Chen and Y. Liu, "Integrated optimization of planning and operations for shared autonomous electric vehicle systems," *Transp. Sci.*, vol. 57, no. 1, pp. 106–134, Jan. 2023, doi: 10.1287/trsc.2022.1156.

[63] F. Maggioni, M. Cagnolari, L. Bertazzi, and S. W. Wallace, "Stochastic optimization models for a bike-sharing problem with transshipment," *Eur. J. Oper. Res.*, vol. 276, no. 1, pp. 272–283, Jul. 2019, doi: 10.1016/j.ejor.2018.12.031.

[64] S. Yan, C.-C. Lu, and M.-H. Wang, "Stochastic fleet deployment models for public bicycle rental systems," *Int. J. Sustain. Transp.*, vol. 12, no. 1, pp. 39–52, Jan. 2018, doi: 10.1080/15568318.2017.1324586.

[65] M. Dell'Amico, M. Iori, S. Novellani, and A. Subramanian, "The bike sharing rebalancing problem with stochastic demands," *Transp. Res. B, Methodol.*, vol. 118, pp. 362–380, Dec. 2018, doi: 10.1016/j.trb.2018.10.015.

[66] R. Regue and W. Recker, "Proactive vehicle routing with inferred demand to solve the bikesharing rebalancing problem," *Transp. Res. E, Logistics Transp. Rev.*, vol. 72, pp. 192–209, Dec. 2014, doi: 10.1016/j.tre.2014.10.005.

[67] D. Huang, X. Chen, Z. Liu, C. Lyu, S. Wang, and X. Chen, "A static bike repositioning model in a hub-and-spoke network framework," *Transp. Res. E, Logistics Transp. Rev.*, vol. 141, Sep. 2020, Art. no. 102031, doi: 10.1016/j.tre.2020.102031.

[68] R. Guo, Z. Jiang, J. Huang, J. Tao, C. Wang, J. Li, and L. Chen, "BikeNet: Accurate bike demand prediction using graph neural networks for station rebalancing," in *Proc. IEEE SmartWorld, Ubiquitous Intell. Comput., Adv. Trusted Comput., Scalable Comput. Commun., Cloud Big Data Computing, Internet People Smart City Innov. (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, Leicester, U.K., 2019, pp. 686–693.

[69] J.-H. Cho, Y.-H. Seo, and D.-K. Kim, "Efficiency comparison of public bike-sharing repositioning strategies based on predicted demand patterns," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2675, no. 11, pp. 104–118, Nov. 2021, doi: 10.1177/03611981211016859.

[70] L. Yu, T. Feng, T. Li, and L. Cheng, "Demand prediction and optimal allocation of shared bikes around urban rail transit stations," *Urban Rail Transit*, vol. 9, no. 1, pp. 57–71, Dec. 2022, doi: 10.1007/s40864-022-00183-w.

[71] Y. Li, W. Y. Szeto, J. Long, and C. S. Shui, "A multiple type bike repositioning problem," *Transp. Res. B, Methodol.*, vol. 90, pp. 263–278, Aug. 2016, doi: 10.1016/j.trb.2016.05.010.

[72] H. Akova, S. Hulagu, and H. B. Celikoglu, "Static bike repositioning problem with heterogeneous distribution characteristics in bike sharing systems," *Transp. Res. Proc.*, vol. 62, pp. 205–212, Jan. 2022, doi: 10.1016/j.trpro.2022.02.026.

[73] T. Raviv, M. Tzur, and I. A. Forma, "Static repositioning in a bike-sharing system: Models and solution approaches," *EURO J. Transp. Logistics*, vol. 2, no. 3, pp. 187–229, Aug. 2013, doi: 10.1007/s13676-012-0017-6.

[74] G. Erdogan, M. Battarra, and R. Wolfler Calvo, "An exact algorithm for the static rebalancing problem arising in bicycle sharing systems," *Eur. J. Oper. Res.*, vol. 245, no. 3, pp. 667–679, Sep. 2015, doi: 10.1016/j.ejor.2015.03.043.

[75] A. A. Kadri, I. Kacem, and K. Labadi, "A branch-and-bound algorithm for solving the static rebalancing problem in bicycle-sharing systems," *Comput. Ind. Eng.*, vol. 95, pp. 41–52, May 2016, doi: 10.1016/j.cie.2016.02.002.

[76] B. P. Bruck, F. Cruz, M. Iori, and A. Subramanian, "The static bike sharing rebalancing problem with forbidden temporary operations," *Transp. Sci.*, vol. 53, no. 3, pp. 882–896, May 2019, doi: 10.1287/trsc.2018.0859.

[77] L. Di Gaspero, A. Rendl, and T. Urli, "A hybrid ACO+CP for balancing bicycle sharing systems," in *Proc. Hybrid Metaheuristics*, vol. 7919. Berlin, Germany: Springer, 2013, pp. 198–212.

[78] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Ann. Stat.*, vol. 29, no. 5, pp. 1189–1232, 2001.

[79] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Nov. 2011.

[80] A. Oyedele, A. Ajayi, L. O. Oyedele, J. M. D. Delgado, L. Akanbi, O. Akinade, H. Owolabi, and M. Bilal, "Deep learning and boosted trees for injuries prediction in power infrastructure projects," *Appl. Soft Comput.*, vol. 110, Oct. 2021, Art. no. 107587, doi: 10.1016/j.asoc.2021.107587.

[81] J. Wu, X.-Y. Chen, H. Zhang, L.-D. Xiong, H. Lei, and S.-H. Deng, "Hyperparameter optimization for machine learning models based on Bayesian optimization," *J. Electron. Sci. Technol.*, vol. 17, pp. 26–40, Mar. 2019, doi: 10.11989/JEST.1674-862X.80904120.

[82] T. Head, G. Louppe, H. Nahrstaedt, I. Shcherbatyi, and M. Kumar. (2020). *Scikit-Optimize*. [Online]. Available: https://github.com/scikit-optimize

[83] *Stata: Time-Series Reference Manual Release 17*, StataCorp LLC, College Station, TX, USA, 2021, pp. 1–52.

[84] M. Dorigo, M. Birattari, and T. Stutzle, "Ant colony optimization," *IEEE Comput. Intell. Mag.*, vol. 1, no. 4, pp. 28–39, Nov. 2006, doi: 10.1109/MCI.2006.329691.

[85] G. Agrover and I. Zidong. (2023). *Scikit-Opt*. [Online]. Available: https://github.com/guofei9987/scikit-opt

[86] W. E. Hart, J.-P. Watson, and D. L. Woodruff, "Pyomo: Modeling and solving mathematical programs in Python," *Math. Program. Comput.*, vol. 3, no. 3, pp. 219–260, Aug. 2011, doi: 10.1007/s12532-011-0026-8.

[87] M. Hua, X. Chen, J. Chen, and Y. Jiang, "Minimizing fleet size and improving vehicle allocation of shared mobility under future uncertainty: A case study of bike sharing," *J. Cleaner Prod.*, vol. 370, Oct. 2022, Art. no. 133434, doi: 10.1016/j.jclepro.2022.133434.

**NARITH SAUM** was born in Cambodia, in 1992. He received the B.Eng. degree in civil engineering from the Institute of Technology of Cambodia, in 2015, and the M.Eng. degree in transportation engineering from Chulalongkorn University, in 2017. He is currently pursuing the dual Ph.D. degree with Hokkaido University and the Sirindhorn International Institute of Technology. His research interests include the implementation of machine learning and deep learning in transportation and traffic network optimization.

**SATOSHI SUGIURA** received the B.E., M.E., and D.Eng. degrees from Gifu University. From 2009 to 2014, he was with Dainichi Consultant Company Ltd. He was with Gifu University as a Research Assistant, from 2014 to 2016, and an Assistant Professor, from 2016 to 2019. In April 2019, he moved to Hokkaido University as an Associate Professor. He has worked on projects related to the maintenance strategy of the infrastructure on the road. His research interests include network analysis using graph theory, linear algebra, and optimization problems.

**MONGKUT PIANTANAKULCHAI** was born in Bangkok, Thailand, in 1973. He received the B.Eng. degree in civil engineering from Chulalongkorn University, in 1994, the M.Eng. degree in transportation engineering from the Asian Institute of Technology, in 1996, and the Ph.D. degree in transportation from Tohoku University, in 2000.

Since 2000, he been with the Sirindhorn International Institute of Technology, Thammasat University, where he is currently an Associate Professor with the School of Civil Engineering and Technology. His recent research interests include intelligent transportation systems, machine learning, and artificial intelligence applications in transportation and logistics.