Transportation Research Part C 131 (2021) 103327

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

The dynamic bike repositioning problem with battery electric vehicles and multiple charging technologies

Yue Wang [a], W.Y. Szeto [b,c,d,*]

[a] School of Management, Huazhong University of Science and Technology, Wuhan, China
[b] Department of Civil Engineering, The University of Hong Kong, Pokfulam Road, Hong Kong
[c] The University of Hong Kong Shenzhen Institute of Research and Innovation, Shenzhen, China
[d] Guangdong–Hong Kong–Macau Joint Laboratory for Smart Cities, China

ARTICLE INFO

Keywords:
Dynamic bike-repositioning problem
Rolling horizon method
Demand forecasting
Battery electric vehicles
Artificial bee colony algorithm

ABSTRACT

The bike-repositioning problem (BRP) primarily involves determining the routes and loading instructions for a fleet of vehicles that transport bikes between stations in a bike sharing system (BSS), to mitigate the mismatch between the demand for and supply of public bikes. However, the use of fossil-fueled vehicles for this repositioning task generates pollutants and greenhouse gases, which harm the environment. The use of battery electric vehicles (BEVs) instead of fossil-fueled vehicles for repositioning bikes can mitigate this negative environmental impact. On the other hand, user demand for bikes is dynamic during the daytime. Therefore, this study addresses the dynamic BRP with battery electric vehicles. Multiple charging technologies are available at charging stations to allow repositioning vehicles to recharge en-route at different speeds and costs. The objective is to solve the problem by minimizing the weighted sum of the penalty costs of unmet user demand and the charging costs of repositioning vehicles. A rolling horizon framework is adopted to incorporate the revealed inventory levels at bike stations, the BEV load of bikes, and the battery energy levels of BEVs at regular intervals. An artificial bee colony algorithm with an embedded dynamic programming method for computing the loading instructions is proposed to generate solutions. Computational experiments are conducted on a real-world BSS, and three aspects of the problem properties are analyzed: the horizon settings, the various penalties for failed rentals and returns, and the charging-related settings. The results provide practical insights into the use of BEVs for daily bike-repositioning tasks in BSSs.

1. Introduction

Bike sharing systems (BSSs) offer bikes for people to rent and are developing rapidly around the world. Some BSSs are station-based systems (e.g., the Bluebikes in Boston, US; Capital Bikeshare in Washington DC, US; and Citybike Wien in Vienna, Austria). Others are dockless (free-floating) systems offered by private companies (McKenzie, 2018), such as Mobike and DiDi Bike in China, and Lime and Spin in the US. The rapid development of BSSs has revealed a common phenomenon in their daily operation: the mismatch in the supply of and demand for bikes, which is caused by asymmetric bike flows. The bike-repositioning problem (BRP) arises from this phenomenon and has attracted attention from operators and researchers in recent years. The BRP involves determining how to best use

* Corresponding author.
E-mail addresses: wyhust@hust.edu.cn (Y. Wang), ceszeto@hku.hk (W.Y. Szeto).

https://doi.org/10.1016/j.trc.2021.103327
Received 11 September 2020; Received in revised form 30 June 2021; Accepted 25 July 2021
Available online 24 August 2021

a fleet of repositioning vehicles to transfer bikes between bike stations to correct the supply and demand mismatch. The solutions to the BRP usually dictate the order in which repositioning vehicles visit bike stations, and how many bikes the vehicles load or unload at each station.

There are two general types of BRP: the static bike-repositioning problem (SBRP) and the dynamic bike-repositioning problem (DBRP). The SBRP assumes that the user demand during the repositioning is negligible or that the repositioning work is conducted when the system is out of service (e.g., during nighttime). Thus, only one decision on repositioning is made, as repositioning does not need to be adapted over time. Most studies have focused on the SBRP (e.g., Erdoğan et al., 2015; Ho and Szeto, 2016; Schuijbroek et al., 2017; Wang and Szeto, 2018; Lahoorpoor et al., 2019). In contrast, the DBRP considers repositioning work that is conducted when the system is in use, i.e., during daytime (e.g., Contardo et al., 2012; Shu et al., 2013; Kloimüllner et al., 2014; Ghosh et al., 2017; Caggiani et al., 2018; Brinkmann et al., 2019). In this scenario, the demand is time-dependent, and the repositioning decisions may need to be modified in response to new information.

The SBRP and the DBRP can be further classified according to possible uncertainty in the forecasted demand. If decisions are made assuming that the demand is certain (uncertain or unknown), then the problem is deterministic (stochastic). Thus, similar to the classification of the vehicle routing problem (Pillac et al., 2013), the BRPs can be classified into four categories: static–deterministic BRPs (e.g., Erdoğan et al., 2015), static–stochastic BRPs (e.g., Schuijbroek et al., 2017), dynamic–deterministic BRPs (e.g., Shui and Szeto, 2018), and dynamic–stochastic BRPs (e.g., Brinkmann et al., 2019). This classification is summarized in Table 1.

The original intention of promoting BSSs was to encourage green and healthy travel. However, repositioning bikes, which is almost a routine operation in BSSs, generates pollutants and greenhouse gases (GHGs), as it uses internal combustion engine vehicles (ICEVs) powered by fossil fuels. Luo et al. (2019) found that the bike-repositioning task was the task that contributed the most to GHG emissions in the lifecycle of BSSs, accounting for 36% and 73% of GHG emissions in station-based and dockless BSSs, respectively.

To mitigate or prevent these health and environmental effects, electric vehicles (EVs) can be used for bike repositioning. EVs can be broadly classified into two types, according to their power source: battery electric vehicles (BEVs), which are powered only by an internal battery, and hybrid EVs (HEVs), which are powered by an internal combustion engine and a battery (Erdelić and Carić, 2019). In this study, we only consider BEVs.

According to the assessment made by Brennan and Barder (2016), BEVs cost less than ICEVs, and the electricity cost per mile for BEVs is lower than the fuel cost per mile for ICEVs. In addition, over its lifecycle, a BEV generates 23% less GHGs than an ICEV. However, BEVs' limited driving range and the uneven distribution of their refueling stations may result in refueling inconvenience. Thus, the scheduling of refueling activities must be incorporated into the planning of BEV-based repositioning tasks.

The use of BEVs for bike repositioning was first proposed by Usama et al. (2019). They assumed that a sole charging station was located at a depot, where all BEVs began and ended their journeys. A mathematical model was developed for the static scenario (i.e., the SBRP) and solved for small networks with a commercial solver. However, the computational time was up to several hours long, and thus their methodology is unsuitable for solving a real-world problem with time-varying demand. In addition, public charging facilities enable repositioning BEVs to recharge en-route, which is more convenient for daily operations than only recharging at a depot. Therefore, the BRP using BEVs can be further extended to daytime repositioning (i.e., the DBRP). Furthermore, multiple charging technologies are available, which have different charging costs and speeds, but this important and practical consideration was not captured by the model of Usama et al. (2019).

In this study, we consider a new DBRP with BEVs and multiple charging technologies (hereafter, the DBRP–BEV). The objective is to minimize the weighted sum of the penalty cost (due to the unmet user demand) and the charging costs of repositioning vehicles. Sequential decisions on vehicle routes, arrival times at visited charging and bike stations, and loading/unloading quantities at visited bike stations are made under a rolling horizon framework to incorporate the revealed inventory levels at bike stations, the load of bikes on a BEV, and the battery energy levels of BEVs at fixed intervals. The artificial bee colony (ABC) algorithm with an embedded dynamic-programming method is proposed and developed to compute the loading instructions and afford solutions. Computational experiments were carried out on a real-world BSS (Capital Bikeshare in Washington, DC), and the results were compared with those from the canonical genetic algorithm (GA) to demonstrate the effectiveness of the proposed method for solving the subproblems. The runtime of the rolling horizon framework is discussed. Three aspects of the problem properties are analyzed: the horizon settings in the rolling horizon scheme, the different penalties for failed rentals and returns, and the charging-related settings (the battery capacity of a BEV, the number of charging stations, and the availability of multiple charging technologies). The results provide practical insights into the use of BEVs for the repositioning task in BSSs.

The contributions of this paper are as follows.

1. We introduce a new DBRP, and propose a rolling horizon framework for the problem and a detailed mathematical model for its subproblems.

Table 1
Classification of bike-repositioning problems (BRPs).

Classification of BRPs	Are repositioning decisions adapted over time?		Is the demand assumed to be certain?		Example
	Yes	No	Yes	No	
Static–deterministic		✓	✓		Erdoğan et al. (2015)
Static–stochastic		✓		✓	Schuijbroek et al. (2017)
Dynamic–deterministic	✓		✓		Shui and Szeto (2018)
Dynamic–stochastic	✓			✓	Brinkmann et al. (2019)

2. We develop a solution method to solve the proposed problem.
3. We provide practical insights into the use of BEVs for daily bike-repositioning tasks in BSSs.

The remainder of the paper is organized as follows. Section 2 reviews the literature on the DBRP and the BEV charging problem. Section 3 describes the DBRP–BEV and its subproblems within a rolling horizon framework. Section 4 illustrates the details of the ABC algorithm used for solving each subproblem, and Section 5 presents computational experiments on a real-world BSS. Section 6 concludes the paper.

2. Literature review

In this section, we first describe the state of the art of the DBRP. Then, we briefly introduce the studies on the charging problem of BEVs.

2.1. The dynamic bike repositioning problem

In the BRP which occurs in the daytime or when the BSS is in use, the time-varying state of the system should be considered when making decisions on vehicle routes and loading/unloading quantities. Under these conditions, the BRP is a dynamic problem (i.e., the DBRP) as the repositioning decisions need to be adjusted over time (Brinkmann et al., 2015). Therefore, we discuss the dynamics of the DBRP in terms of two aspects: the state of a BSS and the repositioning decisions.

The dynamics of the state of a BSS during daytime bike repositioning are two-fold: a time-dependent demand pattern (i.e., user activities) and time-dependent target inventory levels (i.e., repositioning objectives). Some studies on dynamic repositioning have only considered the time-dependent demand pattern, while the target inventory level was assumed to be fixed when planning. For example, Kloimüllner et al. (2014) assumed that the cumulated user demands at bike stations comprised piecewise linear functions of the difference between the expected rentals and returns, based on an hourly discretization of the historical data. They evaluated a repositioning solution according to the unfulfilled demand during the repositioning and the deviation of the final inventory from a fixed target inventory level. Brinkmann et al. (2016) formulated a multi-period problem to handle the time-dependent target inventory levels and demand patterns, but both of these were static when planning the repositioning task in each period. Shui and Szeto (2018) considered time-varying net demand and used rolling planning horizons. However, at each time of planning the repositioning task for a horizon, they fixed the target inventory levels of the bike stations. Some other studies have considered dynamic target-inventory levels. For example, in addition to examining the time-varying demand pattern, Zhang et al. (2017) also captured the changes in the target inventory levels of bike stations depending on the arrival times of vehicles at those stations. In this study, we consider both the time-dependent demand pattern and the time-dependent target inventory levels.

To incorporate the latest information on a BSS and its repositioning vehicles, decisions can be made multiple times and sequentially over the modeling horizon, and the problem can be divided into several subproblems. The rolling horizon method provides a suitable framework for solving such a problem. The concept of a rolling horizon was originally proposed for production planning (Baker, 1977) and involves time-triggered decision points. It is mainly about making periodic planning or decisions incorporating the anticipation of both reliable information in the near future and less reliable information in the slightly less-near future (Peeta and Mahmassani, 1995). Both types of information are used for planning, but the next decision point arises before the end of the current planned period, because the information can be updated. This solution framework breaks the long modeling horizon into short but overlapped periods, thus making a compromise between computational efforts and predictive accuracy. This framework has been applied to address various problems, including the DBRP, and the subproblem for each shorter period can be solved independently by various methods, such as a two-stage (math-) heuristic (e.g., Zhang et al., 2017), the ABC algorithm (e.g., Shui and Szeto, 2018), and a multiobjective evolutionary algorithm based on decomposition (e.g., Hu et al., 2021).

A class of studies has focused on making only one-step decisions at each stop. That is to say, at each decision point, only the next stop and the quantity at the current stop or at the next stop are determined. The decision points are event-, stop-, or location-triggered, in which the event herein means arriving at a stop or location. In this type of planning, the repositioning problem was usually modeled as a Markov Decision Process (MDP) and the objective was to derive a policy to optimize the objective function value. For example, Brinkmann et al. (2015) modeled the short-term repositioning problem based on an MDP, where the decisions were to determine the loading/unloading quantity of bikes at the current bike station and the identity of the next station to visit. The objective was to minimize the expected penalties, and they used a greedy heuristic to derive the policy, in which the bike-loading quantity at the current stop was that which led to a final inventory as close as possible to the target inventory, and the next stop was the unbalanced station that was closest to the current stop. Based on this study, Brinkmann et al. (2019) proposed a dynamic lookahead policy to make online decisions that incorporated the anticipated future demand. Brinkmann et al. (2020) extended the study of Brinkmann et al. (2019) to a multiple-vehicle case. In another example, Li et al. (2018) used the reinforcement learning method to derive the policy that minimized customer loss in the future. Thus, the next stop and the loading/unloading quantity for the vehicle were decided immediately after the current stop was completed, via an offline-learning process that incorporated real-time observations. Legros (2019) also formulated the routing problem as an MDP, and the decisions were the prioritized bike station to visit next and the optimal loading/unloading quantity at that station. The objective of their problem was to minimize the costs induced by unsatisfied demand, and they used an MDP approach to derive the repositioning strategy and a one-step policy improvement method to determine the priority of stations. Their experimental results suggested that the prioritization of bike stations should consider the distance of stations from a given station, the intensity of activity, the inventory level, and the degree of imbalance.

Other methodologies have formulated and solved the DBRP as a single integrated problem. For example, Contardo et al. (2012)

formulated the DBRP as a space–time network flow problem, and solved this problem using the Dantzig–Wolfe and Benders decomposition schemes. Caggiani and Ottomanelli (2012; 2013) formulated the routing problem in the DBRP as a fixed-point nonlinear integer optimization problem and solved this problem using a branch-and-bound procedure. Vogel et al. (2014) proposed a hybrid metaheuristic that combined the large neighborhood search and exact solution techniques to solve a mixed-integer linear programming model. Ghosh et al. (2017) decomposed the original mixed-integer linear programming model into two components using the Lagrangian dual-decomposition technique. Then, they accelerated the solution process by using the abstraction approach to cluster geographically proximal stations.

The aforementioned formulation approaches and solution methods for the DBRP are summarized in Table 2.

In summary, in the DBRP, the modeling horizon with time-varying demand can be decomposed into multiple overlapped periods. Sequential decision points are selected according to how often decisions need to be made and how many steps ahead need to be planned. Finally, an optimization problem is solved, using one of several methods that have been developed, to determine the actions of vehicles at each of these decision points.

2.2. The battery electric vehicle charging problem

EVs have been widely used in passenger and freight transport, which has led to the EV charging problem being studied in various contexts, such as in electric taxis (e.g., Tian et al., 2016), electric car–sharing systems (e.g., Xu and Meng, 2019), electric buses (e.g., Wang et al., 2017), and electric freight vehicles (e.g., Pelletier et al., 2018).

Unlike the conventional vehicle routing problem (VRP) (Dantzig and Ramser, 1959), the EV routing problem (EVRP) must also consider the limited driving range of EVs, as their spent batteries must be replaced with new, charged batteries at swapping stations (e.g., Hof et al., 2017) or recharged at charging stations (e.g., Schneider et al., 2014). Visiting either type of station requires an EV to detour from its route, which adds more complicated time-related constraints that substantially affect the problem formulation, in terms of, for example, the arrival time constraints (e.g., Schneider et al., 2014; Hiermann et al., 2016), or adds a distance-related objective, for example, the minimization of the total travel distance (e.g., Keskin and Çatay, 2016). Therefore, when and where to exchange or recharge a battery must be carefully determined.

There are many variants of the VRP that address the problem of detouring to refueling stations, including but not limited to swapping or charging stations. For example, Erdoğan and Miller-Hooks (2012) proposed a green VRP in which vehicles were powered by alternative fuels and detoured to fueling stations that were limited in number and unevenly distributed. Schneider et al. (2015) considered a more general VRP with intermediate stops for different purposes, such as replenishment, unloading operations, and refueling. Schiffer and Walther (2017) simultaneously considered the siting of charging stations and the routing problem of EVs as a location-routing problem. Hof et al. (2017) also considered a location-routing problem but used battery-swapping stations instead of charging stations. Some studies have also considered vehicles that recharged at customers' locations (e.g., Conrad and Figliozzi (2011) and Schiffer and Walther (2017)).

The diversity of existing charging facilities means that more than one charging technology can be used. For example, Felipe et al. (2014) solved the EVRP by considering multiple charging technologies, namely slow, medium, fast, and ultrafast technologies with varying charging speeds and costs, and compared the performance of various configurations of these technologies. Their experimental results showed that charging costs were reduced significantly when multiple charging technologies were available. In addition, the advantages of partial recharge were smaller when considering multiple charging technologies because of the higher flexibility of recharging. Montoya et al. (2017) considered three types of charging stations, with each type having a single charging technology; similar settings were used by Macrina et al. (2019). Keskin and Çatay (2018) studied normal, fast, and super-fast charging technologies with different charging speeds and costs. They concluded that the use of fast charging led to reductions in fleet size and energy consumption. Erdelić and Carić (2019) summarized various charging technologies with different power ratings and thus different full-cycle charging times as follows. The power rating of a slow charge was 3 kW, with a charging time of 6–8 h, that of a fast charge was 7–42 kW with a charging time of 1–2 h, and that of a rapid charge was 50–250 kW with a charging time of 5–30 min.

Previously reported solution methods for the EVRP can be broadly classified as exact methods and heuristics. An example of an exact method is the branch-price-and-cut algorithm based on the labeling algorithm, as reported by Desaulniers et al. (2016). This

Table 2
Summary of formulation approaches and solution methods for the DBRP.

Problem formulation approach		Reference	Solution method
Multiple subproblems	Time-triggered	Zhang et al. (2017)	Two-stage (math-) heuristic
		Shui and Szeto (2018)	Artificial bee colony algorithm
		Hu et al. (2021)	Multiobjective evolutionary algorithm based on decomposition
	Event-triggered	Brinkmann et al. (2015)	Greedy heuristic
		Li et al. (2018)	Reinforcement learning
		Legros (2019)	Markov decision process and a one-step policy improvement
Single integrated problem		Contardo et al. (2012)	Dantzig–Wolfe and Benders decomposition
		Caggiani and Ottomanelli (2012; 2013)	Branch-and-bound
		Vogel et al. (2014)	Large neighborhood search and exact solution techniques
		Ghosh et al. (2017)	Lagrangian dual decomposition

method was used to solve problems involving 100 customers and 21 charging stations. Examples of heuristic approaches are two construction heuristics combined with an improvement procedure (e.g., Erdoğan and Miller-Hooks, 2012), a non-deterministic simulated annealing algorithm (e.g., Felipe et al., 2014), the adaptive variable neighborhood search algorithm (e.g., Schneider et al., 2015), the iterated variable neighborhood search (e.g., Lu et al., 2020), the adaptive large neighborhood search (e.g., Hiermann et al., 2016; Keskin and Çatay, 2018; Keskin et al., 2021), and the hybrid iterated local search (e.g., Montoya et al., 2017). The advantage of heuristics is its ability to provide good solutions within a short computation time.

The solution methods for the EVRP are summarized in Table 3. For more information on variants of and solution methods for the EVRP, readers may refer to Erdelić and Carić (2019).

The DBRP–BEV in this study differs from the conventional EVRP in the following two aspects. First, the conventional EVRP usually considers a route-duration limit such that a vehicle can choose to recharge en-route by only the amount that enables it to return to its depot (Montoya et al., 2017). However, the DBRP–BEV considers a 24/7 dynamic system such that vehicles have open routes, without consideration of a depot (Savelsbergh and Sol, 1995). Second, the customer demand is fixed in the EVRP, whereas the DBRP–BEV also has decision variables for the loading/unloading quantities at each stop. This study is the first to consider the use of BEVs in the DBRP.

3. The dynamic bike repositioning problem with battery electric vehicles

In Section 3.1, we describe the DBRP–BEV as a master problem, and in Section 3.2, we introduce the rolling horizon framework to solve this problem. In Section 3.3, we define and formulate the subproblems. Finally, in Section 3.4, we illustrate the connections between the subproblems.

3.1. Description of the master problem

This study considers the practice of using BEVs for repositioning bikes in BSSs. The network consists of a set of bike stations S in a BSS, a set of charging stations C for BEVs, and a set of links. Links are used to connect stations (or nodes) in the network. For each bike station $s \in S(s = 1,2,\cdots,|S|)$, the capacity for bikes (i.e., the number of lockers) is c_s. However, it is assumed that each charging station has no capacity constraint. The locations of bike and charging stations are known and fixed over time, and a charging station may be equipped with one or multiple charging technologies. To simplify the formulation, we assume that a charging station with multiple technologies is modeled by multiple nodes, each of which has a single charging technology. In addition, to allow for multiple visits by each vehicle to a charging station in C, a set C' of dummy nodes is defined such that a visit to one dummy node corresponds to a visit to one charging station with one charging technology, and the number of visits to each dummy node is not greater than one.

A sample network is used to depict the expanded network, as shown in Fig. 1, in which links are omitted for clarity. Fig. 1(a) is the original network, composed of one charging station (i.e., charging node) and four bike stations. Fig. 1(b) is a semi-expanded network that contains an expansion of the charging node to three dummy nodes to represent the three different charging technologies at the charging station. Each dummy node stands for a single visit to one charging technology at the charging station. Fig. 1(c) is a fully expanded network that allows for multiple visits to the charging station, as it contains multiple copies of the dummy nodes, each of which stands for a single visit to a charging station with one charging technology.

The arrival of renters (returners) at a bike station follows a Poisson process, and they rent (return) bikes with a time-varying rate. A failed rental occurs when a customer who wishes to rent a bike arrives at an empty bike station, while a failed return occurs when a customer who wishes to return a bike arrives at a full bike station. A fleet of identical BEVs is denoted set V, and each BEV has a capacity Q for bikes and a battery capacity VB (in kWh). The capacitated BEVs travel between bike stations over time to redistribute bikes by pick-up or delivery such that failed rentals and returns in the immediate future are reduced. We consider batch loading and unloading, and hence the operational time at bike stations is constant and can be integrated into the travel time of a BEV. The travel time between the nodes, together with the integrated loading/unloading time, is expressed as a matrix with elements $t_{i,j}$ (in min), where $i,j \in S \cup C$'.

In addition to planning the visiting sequence of BEVs to bike stations and BEVs' loading/unloading quantities and arrival time at each station, this problem involves scheduling BEVs' recharging activities, i.e.*, when and where a BEV is recharged.* The battery of each BEV is

Table 3
Summary of solution methods for the EVRP.

Solution method		References
Exact method	Branch-price-and-cut algorithm	Desaulniers et al. (2016)
Heuristics	Modified Clarke and Wright savings heuristic, density-based clustering algorithm, and a customized improvement technique	Erdoğan and Miller-Hooks (2012)
	Simulated annealing	Felipe et al. (2014)
	Adaptive variable neighborhood search	Schneider et al. (2015)
	Iterated variable neighborhood search	Lu et al. (2020)
	Adaptive large neighborhood search	Hiermann et al. (2016) Keskin and Çatay (2018) Keskin et al. (2021)
	Hybrid iterated local search	Montoya et al. (2017)

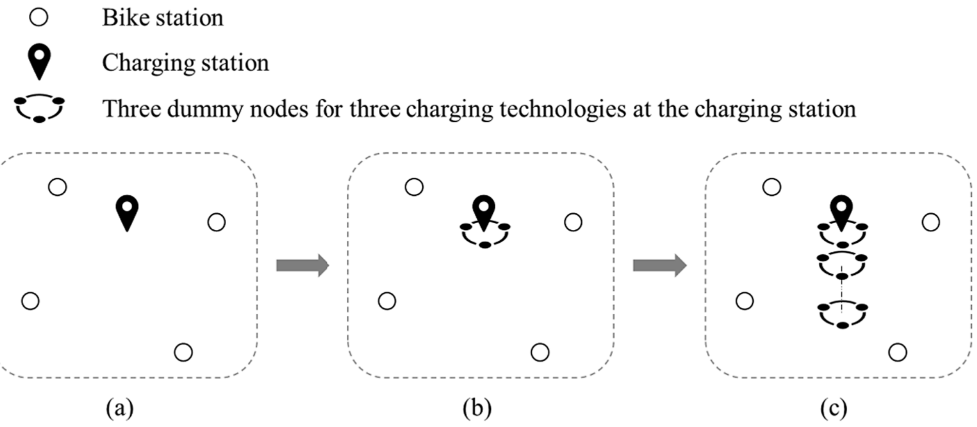

Bike station
Charging station
Three dummy nodes for three charging technologies at the charging station
(a)
(b)
(c)

Fig. 1. Expansion of a sample network: (a) original network; (b) semi-expanded network; (c) fully expanded network.

assumed to be fully charged whenever it leaves a charging station. The state of charge (SOC, in kWh) of a BEV decreases at a constant rate u over its working time (i.e., in kWh/min); accordingly, battery autonomy is defined as VB/u (in min). The SOC of a BEV ranges from 0 to VB. The charging speed of a charging technology at node $i \in C'$ is denoted as g_i (in kWh/min); thus, the full-cycle charging time is VB/g_i (in min). Charging facilities are assumed to be sufficient such that a BEV does not need to queue at a charging station. Therefore, the duration of time that a BEV remains at a charging station (in min) is proportional to the difference between the VB and the SOC of the BEV when it arrives at the charging station. *The repositioning plan and recharging schedule are revised at regular intervals, according to the updated information received from the system.*

The abovementioned master problem is addressed within a rolling horizon framework, which is described as follows.

3.2. Rolling horizon framework for the master problem

We use the rolling horizon method to manage the dynamics of the master problem, as shown in Fig. 2. We consider a modeling horizon that can be infinitely long. Over this horizon, the system status—which comprises the bike station status (the inventory level at all bike stations) and the repositioning BEV status (the BEV location, its load of bikes, and its SOC)—is revealed at every fixed interval t_r (in min). This is called the *rolling step* or the *planning interval*, and is shorter than *the planning horizon.* At each time of planning $w^0 - t_c$ (in min), the BEV routes, the BEV loading/unloading instructions at visited bike stations, and the BEV arrival times at visited bike and charging stations in the *planning horizon* are computed. This determines the performance measures, namely the charging cost in the planning horizon and the penalties for unmet demand for bikes and lockers in the *forecasting horizon.* The forecasting horizon is always longer than the planning horizon to ensure that the penalties for unmet demand that occur in sufficient time after repositioning can be determined. The problem in each forecasting horizon is denoted a subproblem. The time allowed for solving a subproblem is t_c min.

We take the forecasting horizon P2 (i.e., $[w^0, w^f]$) in Fig. 2 as an example, and aim to determine the routes, loading/unloading quantities, and recharging schedules for all of the BEVs during the planning horizon $[w^0, w^p]$ such that the weighted sum of the penalty costs for unmet demand at all bike stations during the forecasting horizon $[w^0, w^f]$ and the charging costs of all the vehicles during the planning horizon $[w^0, w^p]$ is minimized. As the calculation takes t_c min, the realized statuses of bike stations and BEVs must be revealed at time $w^0 - t_c$. During the time interval $[w^0 - t_c, w^0]$, the BEVs continue to follow the most recent instructions that were computed based on the forecasting horizon P1. However, from time w^0 onwards, the BEVs follow the newly generated instructions based on the

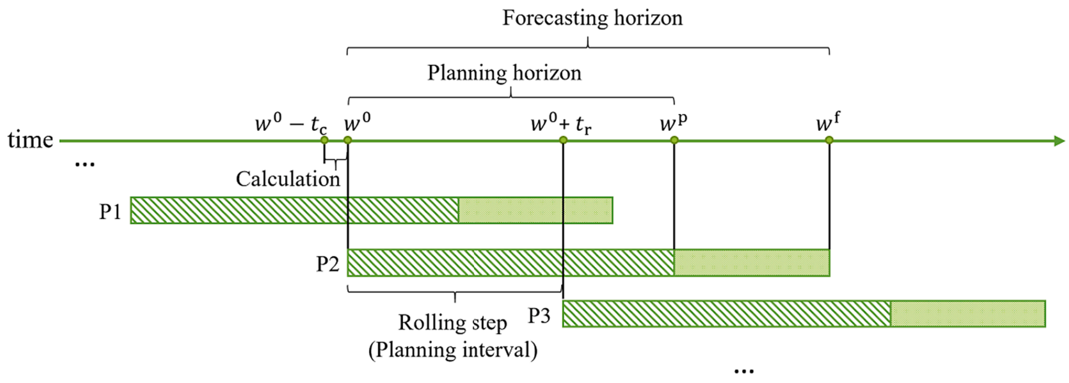


Fig. 2. Rolling horizon framework for the dynamic bike-repositioning problem.

forecasting horizon P2, until the start of the next forecasting horizon (P3) at time $w^0 + t_r$. In the following section, we focus primarily on the subproblem for a single forecasting horizon, P2.

3.3. *Subproblem for a single forecasting horizon*

In this section, we aim to define the components of the subproblem for a single forecasting horizon in general, namely, the estimation of expected unmet demand for bikes and lockers (Section 3.3.1), the decision variables (Section 3.3.2), the objective function (Section 3.3.3), and the constraints (Section 3.3.4).

3.3.1. *Expected unmet demand for bikes and lockers*

The Poisson process has been widely used to describe the arrivals of renters and returners in a BSS (e.g., Raviv and Kolka, 2013; Alvarez-Valdes et al., 2016; Schuijbroek et al., 2017; Zhang et al., 2017; Chiariotti et al., 2018; Datner et al., 2019; Yi et al., 2019). Based on the assumption of Poisson processes of the arrivals of renters and returners, the evolution of inventory levels has been modeled as a Markov chain (Ross, 2014). The expected unmet demand during a forecasting horizon is estimated by discretizing the horizon into multiple short periods. The transition probability within each period is calculated using different approaches, such as the approximation method (e.g., Raviv and Kolka, 2013), which was extended to accommodate the variable starting time and forecasting horizon length in the DBRP (Zhang et al., 2017). This transition probability is then used to calculate the expected values of satisfied/unsatisfied demand for a given initial inventory.

As shown in Fig. 2, the interval for the forecasting horizon P2 is $[w^0, w^{\mathrm{f}}]$ $(w^{\mathrm{f}} \geq w^{\mathrm{p}})$, during which time the unmet demands for bikes and lockers are estimated for use as measures in the objective function. To precisely define these measures, the following notations are adopted. The route of a BEV $v \in V$ $(v = 1, 2, \cdots, |V|)$ is denoted as $RT_v = \{r_1^v, r_2^v, \cdots, r_{|RT_v|}^v\}$, where $r_a^v \in S \cup C^{'}$ $(a = 1, \cdots, |RT_v|)$ and $(\bigcap_{v \in V} RT_v) \cap S = \varnothing$, i.e., each bike station is visited no more than once by all of the BEVs during the planning horizon. The arrival time at bike station r_a^v is $\iota_{r_a^v}$ (in min), where $w^0 \leq \iota_{r_a^v} \leq w^{\mathrm{p}}$. The loading instruction of BEV v is denoted $LQ_v = \{lq_{r_1^v}, lq_{r_2^v}, \cdots lq_{r_{|RT_v|}^v}\}$, where the absolute value of the element $lq_{r_a^v}$ $(a = 1, \cdots, |RT_v|)$ indicates the quantity involved and is positive for a loading action and negative for an unloading action. The load of bikes on the BEV must not exceed its capacity Q (in terms of bikes). The SOC of BEV $v \in V$ must be within the range $[0, VB]$. We define the set of visited bike stations $\widehat{N} = \{s | s \in S \cap (\bigcup_{v \in V} RT_v)\}$, and the set of unvisited bike stations $\breve{N} = S - \widehat{N}$. We use $I_{w^0}^s$ (in terms of bikes) to denote the inventory of bike station s at time w^0.

If $s \in \breve{N}$, the expected unmet demand for bikes during the forecasting horizon $[w^0, w^{\mathrm{f}}]$ is $EB(I_{w^0}^s, w^0, w^{\mathrm{f}})$ (in terms of bikes) and for lockers is $EL(I_{w^0}^s, w^0, w^{\mathrm{f}})$ (in terms of lockers). If $s \in \widehat{N}$, the expected unmet demand for bikes and for lockers before the arrival time ι_s at bike station s (i.e., during the period $[w^0, \iota_s]$) is $EB(I_{w^0}^s, w^0, \iota_s)$ and $EL(I_{w^0}^s, w^0, \iota_s)$, respectively. The expected inventory level at bike station s upon the arrival of a BEV (i.e., at time ι_s) is $EI(I_{w^0}^s, w^0, \iota_s)$ bikes, so the station inventory upon the departure of the BEV is $I_{\iota_s}^s = EI(I_{w^0}^s, w^0, \iota_s) - lq_s$ bikes, where lq_s is the quantity of bikes loaded/unloaded at s (and is positive for a loading action and negative for an unloading action). Then, the unmet demand for bikes and lockers after visiting time ι_s (i.e., during the period $[\iota_s, w^{\mathrm{f}}]$) is $EB\left(I_{\iota_s}^s, \iota_s, w^{\mathrm{f}}\right)$ (in terms of bikes) and $EL\left(I_{\iota_s}^s, \iota_s, w^{\mathrm{f}}\right)$ (in terms of lockers), respectively. Therefore, if $s \in \widehat{N}$, the total unmet demand for bikes in the forecasting horizon $[w^0, w^{\mathrm{f}}]$ is $EB(I_{w^0}^s, w^0, \iota_s) + EB\left(I_{\iota_s}^s, \iota_s, w^{\mathrm{f}}\right)$, and that for lockers is $EL(I_{w^0}^s, w^0, \iota_s) + EL\left(I_{\iota_s}^s, \iota_s, w^{\mathrm{f}}\right)$.

We now explain the estimation of the function values mentioned above, i.e., $EB(\bullet, \bullet, \bullet)$, $EL(\bullet, \bullet, \bullet)$, and $EI(\bullet, \bullet, \bullet)$. The subproblem considers a time-dependent demand pattern and time-dependent target inventory levels. We assume that the arrivals of renters and returners follow two independent non-time-homogenous Poisson processes. We follow the method proposed by Raviv and Kolka (2013) and use a continuous-time Markov chain to depict the dynamics of the station state (i.e., the inventory level). We define the following notations for a general bike station to elaborate on the forecasting procedure.

μ_t	Arrival rate of renters at time t (renters/min)
λ_t	Arrival rate of returners at time t (returners/min)
$\pi_{p,q}(t)$	Transition probability that the inventory level of the station varies from p bikes at the beginning to q bikes at time t
$\boldsymbol{\pi}(t)$	Matrix of transition probabilities $\pi_{p,q}(t)$ from the beginning to time t

As shown in Fig. 3, the station inventory is assumed to be I_1 bikes at time t_1. Thus, the expected unmet demands for bikes and lockers between t_1 and t_2 can be respectively defined as

$$EB(I_1, t_1, t_2) = \int_{t_1}^{t_2} \pi_{I_1,0}(\tau)\mu_\tau \mathrm{d}\tau \tag{3.1}$$

and

$$EL(I_1, t_1, t_2) = \int_{t_1}^{t_2} \pi_{I_1,C}(\tau)\lambda_\tau \mathrm{d}\tau. \tag{3.2}$$

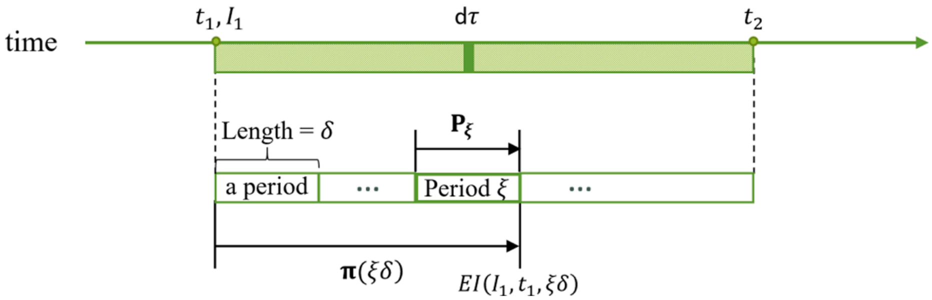


Fig. 3. Forecasts of unmet demands and expected inventory level.

The transition probability matrix $\boldsymbol{\pi}(t)$ is approximated by dividing the forecasting horizon into several continuous periods, each of which has length δ (in min). We assume that t_1 and t_2 are both multiples of δ, and that during a specific period ξ, the arrival rates of renters μ_ξ and returners λ_ξ are constant. μ_ξ and λ_ξ are estimated from historical data (e.g., Kaspi et al., 2014; Chiariotti et al., 2018), and the transition matrix in period ξ is estimated from

$$\mathbf{P}_\xi = e^{\mathbf{R}_\xi \bullet \delta} = \lim_{M\to\infty}\left(\mathbf{I} + \frac{\mathbf{R}_\xi \delta}{M}\right)^M, \tag{3.3}$$

where I is an identity matrix, M is a large constant, and $\mathbf{R}_\xi$ is the transition rate matrix in period ξ:

$$\mathbf{R}_\xi = \begin{bmatrix} -\lambda_\xi & \lambda_\xi & 0 & 0 & \\ \mu_\xi & -(\mu_\xi+\lambda_\xi) & \lambda_\xi & 0 & \cdots \\ 0 & \mu_\xi & -(\mu_\xi+\lambda_\xi) & \lambda_\xi & \\ & \vdots & & \ddots & \vdots \\ & \cdots & & \mu_\xi & -\mu_\xi \end{bmatrix} \tag{3.4}$$

Then, after we derive the transition matrix in each period, the transition probability matrix $\boldsymbol{\pi}(t)$ is calculated as

$$\boldsymbol{\pi}(\xi\delta) = \prod_{\xi'=1}^{\xi} \mathbf{P}_{\xi'} = \boldsymbol{\pi}\langle(\xi-1)\delta\rangle \bullet \mathbf{P}_\xi \tag{3.5}$$

We approximate the expected unmet demands for bikes and lockers in equations (3.1) and (3.2), respectively, as

$$EB(I_1, t_1, t_2) \approx \delta \sum_{\xi=1}^{(t_2-t_1)/\delta} \pi_{I_1,0}(\xi\delta)\mu_\xi \tag{3.6}$$

and

$$EL(I_1, t_1, t_2) \approx \delta \sum_{\xi=1}^{(t_2-t_1)/\delta} \pi_{I_1,C}(\xi\delta)\lambda_\xi. \tag{3.7}$$

The expected inventory level at time t_2 is correspondingly estimated as

$$EI(I_1, t_1, t_2) = \sum_{I'=0}^{C} \pi_{I_1,I'}(t_2) \bullet \mathrm{I}'. \tag{3.8}$$

3.3.2. Decision variables

To clearly define the decision variables, the objective function in Section 3.3.3, and the constraints in Section 3.3.4, the following set notations are introduced:

C'	The set of dummy nodes for modeling multiple visits to the charging stations in set C with multiple charging technologies, where a visit to one dummy node corresponds to a visit to one charging station with one charging technology
N^+	All of the nodes in the sets S and C', i.e.,$N^+ = S \cup C'$
N_σ	The set of the starting locations (or source nodes) of BEVs, i.e., the first stop of each BEV during the planning horizon $[w^0, w^p]$, which is known when planning and where$N_\sigma \subset N^+$
N_τ	The set of dummy sink nodes that represent the final locations of each BEV, $\lvert N_\tau\rvert = \lvert V\rvert$. (As the final visit node is unknown and a BEV does not always return to the depot at the end of each planning horizon, introducing dummy sink nodes allows us to model BEV routes using traditional flow conservation constraints.)
$\overline{N}$	All of the nodes in the set N^+ but not in the set N_σ, i.e.,$\overline{N} = N^+ - N_\sigma$
N^-	All of the nodes in the sets $\overline{N}$ and N_τ, i.e.,$N^- = \overline{N} \cup N_\tau$
N^*	All of the nodes in the sets N_σ, $\overline{N}$, and N_τ, i.e.,$N^* = N_\sigma \cup \overline{N} \cup N_\tau$

The decision variables comprise the following:

x_{ijv}	$=\begin{cases}1, & \text{if BEV } v \text{ travels directly from node } i \text{ to node } j\\ 0, & \text{otherwise}\end{cases}, \forall i \in N^{+}, \forall j \in N^{-}, i \neq j, \forall v \in V$
f_{ijv}	Number of bikes on BEV v when it travels directly from node i to node j, $\forall i \in N^{+}, \forall j \in N^{-}, i \neq j, \forall v \in V$ (in terms of bikes)
z_i	$=\begin{cases}1, & \text{if node } i \text{ is visited}\\ 0, & \text{otherwise}\end{cases}, \forall i \in N^{+}$
lq_i	Loading/unloading quantity of a BEV at bike station $i \in S$ (in terms of bikes)
b_i	Remaining battery capacity of a BEV upon its arrival at node $i \in N^{+}$ (in kWh)
ι_i	Arrival time of a BEV at node $i \in N^{+}$ (in min)

3.3.3. Objective function

In each subproblem, the forecasted unmet demand of all bike stations during the corresponding forecasting horizon is considered. If a penalty β (in \$) is assigned to a failed rental and a penalty γ (in \$) is assigned to a failed return, then the total penalty costs (in \$) for the unmet demand of all of the bike stations in the system are

$$\begin{aligned}P = \beta\sum_{i \in S}\Big(\big(EB\big(I^{i}_{w^0}, w^0, \iota_i\big) + EB\big(EI\big(I^{i}_{w^0}, w^0, \iota_i\big) - lq_i, \iota_i, w^{\mathrm{f}}\big)\big) \cdot z_i + EB\big(I^{i}_{w^0}, w^0, w^{\mathrm{f}}\big) \cdot (1 - z_i)\Big) + \gamma\sum_{i \in S}\Big(\big(EL\big(I^{i}_{w^0}, w^0, \iota_i\big)\\ + EL\big(EI\big(I^{i}_{w^0}, w^0, \iota_i\big) - lq_i, \iota_i, w^{\mathrm{f}}\big)\big) \cdot z_i + EL\big(I^{i}_{w^0}, w^0, w^{\mathrm{f}}\big) \cdot (1 - z_i)\Big).\end{aligned} \tag{3.9}$$

The charging cost is composed of a fixed cost cr_0 (in \$) and a variable cost for each charge, which is the product of the unit charging cost cr_i (in \$/min) and the charging time $(VB - b_i)/g_i$ (in min) at node $i \in C^{'}$. The sum of the charging costs en-route is multiplied by a weight α, and the result is added to the objective function. Therefore, the objective of a subproblem of the DBRP–BEV is

$$\text{Minimize} P + \alpha \cdot \sum_{i \in C^{'}} z_i \cdot \left(cr_0 + cr_i \cdot \frac{VB - b_i}{g_i}\right). \tag{3.10}$$

3.3.4. Constraints

The following constraints are applied:

$$\sum_{v \in V}\sum_{j \in N^{-}, j \neq i} x_{ijv} = 1, \forall i \in N_{\sigma} \tag{3.11}$$

$$\sum_{v \in V}\sum_{i \in N^{+}, i \neq j} x_{ijv} = 1, \forall j \in N_{\tau} \tag{3.12}$$

$$\sum_{v \in V}\sum_{j \in N^{-}, j \neq i} x_{ijv} \leq 1, \forall i \in \overline{N} \tag{3.13}$$

$$\sum_{v \in V}\sum_{j \in N^{-}, j \neq i} x_{ijv} - z_i = 0, \forall i \in N^{+} \tag{3.14}$$

$$\sum_{j \in N^{+}, j \neq i} x_{jiv} - \sum_{j \in N^{-}, j \neq i} x_{ijv} = 0, \forall i \in \overline{N}, \forall v \in V \tag{3.15}$$

$$0 \leq f_{ijv} \leq Q \cdot x_{ijv}, \forall i \in N^{+}, \forall j \in N^{-}, i \neq j, \forall v \in V \tag{3.16}$$

$$\sum_{v \in V}\sum_{j \in N^{-}, j \neq i} f_{ijv} = \sum_{v \in V}\sum_{j \in N^{+}, j \neq i} f_{jiv}, \forall i \in C^{'} \tag{3.17}$$

$$lq_i = \sum_{v \in V}\sum_{j \in N^{-}, j \neq i} f_{ijv} - \sum_{v \in V}\sum_{j \in N^{+}, j \neq i} f_{jiv}, \forall i \in S \tag{3.18}$$

$$-c_i \leq lq_i \leq c_i, \forall i \in S \tag{3.19}$$

$$0 \leq EI\big(I^{i}_{w^0}, w^0, \iota_i\big) - lq_i \leq c_i, \forall i \in S \tag{3.20}$$

$$\iota_i + t_{ij} \cdot x_{ijv} - w^{\mathrm{p}} \cdot \big(1 - x_{ijv}\big) \leq \iota_j, \forall i \in S, \forall j \in \overline{N}, i \neq j, \forall v \in V \tag{3.21}$$

$$\iota_i + t_{ij} \cdot x_{ijv} + \frac{VB - b_i}{g_i} - \left(w^{\mathrm{p}} + \frac{VB}{g_i}\right) \cdot \big(1 - x_{ijv}\big) \leq \iota_j, \forall i \in C^{'}, \forall j \in \overline{N}, i \neq j, \forall v \in V \tag{3.22}$$

$$w^0 \leq \iota_i \leq w^0 \cdot (1 - z_i) + w^{\mathrm{p}} \cdot z_i, \forall i \in N^{+} \tag{3.23}$$

$$b_j \leq b_i - \big(u \cdot t_{ij}\big) \cdot x_{ijv} + VB \cdot \big(1 - x_{ijv}\big), \forall i \in S, \forall j \in \overline{N}, i \neq j, \forall v \in V \tag{3.24}$$

$$b_j \leq VB - \left(u \bullet t_{ij}\right) \cdot x_{ijv}, \forall i \in C^{'}, \forall j \in \overline{N}, i \neq j, \forall v \in V \tag{3.25}$$

$$b_i \geq u \bullet \mathrm{min}_{\mathrm{j} \in \mathrm{C}'} \mathrm{t}_{\mathrm{ij}}, \forall \mathrm{i} \in \mathrm{S} \tag{3.26}$$

$$-Q \leq lq_i \leq Q, \forall i \in S \tag{3.27}$$

$$0 \leq b_i \leq VB, \forall i \in N^+ \tag{3.28}$$

$$w^0 \leq \iota_i \leq w^{\mathrm{p}}, \forall i \in N^+ \tag{3.29}$$

$$x_{ijv} \in \{0,1\}, \forall i \in N^+, \forall j \in N^-, i \neq j, \forall v \in V \tag{3.30}$$

$$z_i \in \{0,1\}, \forall i \in N^+ \tag{3.31}$$

The objective (3.10) minimizes the weighted sum of the penalty cost for the unmet demand and the charging cost for the BEVs. Constraints (3.11) and (3.12) restrict the starting locations and end locations of the BEVs. Constraints (3.13) stipulate that each node can be visited no more than once by all BEVs. Constraints (3.14) define the variables z_i for each node in the set N^+. Constraints (3.15) are vehicle flow conservation equations. Constraints (3.16) set the range of BEV loads. Constraints (3.17) ensure that there are no loading/unloading actions at charging stations. Constraints (3.18) define the loading/unloading quantity of a BEV at bike stations, and Constraints (3.19) limit this quantity to no greater than the station capacity. Constraints (3.20) ensure that the bike station inventory after being visited by a BEV does not exceed the station capacity. The function $EI\left(I^i_{w^0}, w^0, \iota_i\right)$ is defined in Equation (3.8). Constraints (3.21) and (3.22) ensure the feasibility of time after a BEV visits a bike station and a charging station, respectively. Besides the travel time between two nodes, the arrival time at the next node after leaving the charging station should additionally consider the charging time. Constraints (3.23) set the range of the BEV arrival time at each node for the set N^+. Constraints (3.24) and (3.25) stipulate that the SOC of a BEV decreases in proportion to its travel time and that a BEV is fully recharged when it leaves a charging station. Constraints (3.26) ensure that the SOC of a BEV always supports it visiting the nearest charging station. Constraints (3.27)–(3.31) are domain constraints. This subproblem is formulated as a mixed-integer nonlinear programming problem and is NP-hard.

3.4. *Connections between subproblems*

The solution to the subproblem described above guides the repositioning task for the corresponding horizon. Two adjacent subproblems are connected via the revealed states of bike stations and BEVs at the time of planning, which are updated by the simulation of user activities and repositioning tasks during the rolling step. However, due to the discrepancy between the forecasted inventory level and the revealed value, the planned loading/unloading quantity may not be realized. Therefore, the loading/unloading quantities must adapt to reality during the simulation.

In the simulation, a repositioning BEV arrives at a bike station s, where it loads/unloads a greedy number of bikes that is closest in value to the planned quantity lq_s. Specifically, if $lq_s > 0$ (i.e., the plan is to load lq_s bikes at bike station s), then the realized amount is the minimum value of the realized inventory level at bike station s, the realized remaining capacity of the BEV, and the planned quantity lq_s. If $lq_s < 0$ (i.e., the plan is to unload $-lq_s$ bikes at bike station s), then the realized amount is the minimum value of the realized remaining capacity of bike station s, the realized BEV load, and the planned quantity $-lq_s$. The load of a BEV and the inventory level at a bike station are updated according to the realized amounts. If a customer wishing to rent a bike arrives at an empty station, the count of failed rentals is increased by one, and the station remains empty. If a customer wishing to return a bike arrives at a full station, the count of failed returns is increased by one, and the station remains full.

Under this rule, the simulated BSS evolves with all of the realized rental/return events and repositioning actions in the rolling step $[w^0, w^0 + t_{\mathrm{r}}]$. At time $w^0 + t_{\mathrm{r}} - t_{\mathrm{c}}$, the instantaneous status of bike stations and repositioning BEVs are revealed for the next subproblem of the forecasting horizon P3, and new instructions are computed. From time $w^0 + t_{\mathrm{r}}$ onwards, the BEVs follow the newly generated instructions, which are based on P3. Thus, the horizon P2 rolls to the next.

4. Use of the artificial bee colony algorithm for solving subproblems

The ABC algorithm is selected to solve subproblems, as it has proven highly suitable for performing combinatorial optimization, such as in solving the traveling salesman problem (e.g., Karaboga and Gorkemli, 2011) and the capacitated vehicle routing problem (e.g., Szeto et al., 2011). It has also proven highly effective for solving the SBRP (e.g., Wang and Szeto, 2021) and the DBRP (e.g., Shui and Szeto, 2018). Thus, we first introduce the overall framework of the ABC algorithm in Section 4.1. Then, we describe the details of the ABC algorithm implemented in this study, namely the solution representation (Section 4.2), the route initialization (Section 4.3), the computation of the loading instructions (Section 4.4), the evaluation of a solution (Section 4.5), and the search frameworks of three types of bees (Section 4.6).

4.1. *Overall framework of the artificial bee colony algorithm*

The ABC algorithm (Karaboga, 2005) is a population-based heuristic that mimics the food-searching behavior of bees. It uses three main types of bees—employed bees, onlooker bees, and scout bees—which represent three different search frameworks. Thus, employed bees exploit each of a given set of food sources (i.e., solutions) to obtain improvements, onlooker bees prefer to explore promising areas, and scout bees explore a new food source if a certain food source has limited potential to be improved.

The framework of the ABC algorithm is summarized in **Algorithm 1**. A set of solutions $F = \{x_1, x_2, \cdots x_{|F|}\}$ are generated at the beginning, and each solution x_m ($m = 1, \cdots, |F|$) is associated with a count l_m of unsuccessful attempts at improvement (i.e., a neighborhood search that fails to find a better solution). Each solution is evaluated to obtain a fitness value based on the objective function, where a higher fitness represents a better solution. The fitness function ***f(•)*** for a minimization problem is defined as the reciprocal of the objective function. Then, in the **while** loop (Line 3 to Line 25 in **Algorithm 1**), each solution x_m is first searched by an employed bee in the neighborhood (the subroutine *EmployedBee*(•) in Line 5 of **Algorithm 1**). If the employed bee finds a better solution $\widetilde{x}_m$, this replaces the original solution x_m, and the count l_m of this solution is reset to 0. Otherwise, l_m is increased by one. Then, solution $\widehat{x}_m$ is initialized with the incumbent solution x_m to prepare for the onlooker bee phase. During the onlooker bee phase, each solution x_m is selected with a probability proportional to its fitness value. If the onlooker bee (the subroutine *OnlookerBee*(•) in Line 12 of **Algorithm 1**) finds solution $\widetilde{x}_m$, and this is better than the associated $\widehat{x}_m$, then $\widehat{x}_m$ is updated to $\widetilde{x}_m$. At the end of the onlooker bee phase, if $\widehat{x}_m$ is an improved solution (i.e., $f\left(\widehat{x}_m\right) > f(x_m)$), solution x_m is replaced with $\widehat{x}_m$ and l_m is reset to 0; otherwise, l_m is increased by one. If the value of l_m exceeds a threshold *limit*, then x_m is replaced with a new solution generated by a scout bee (the subroutine *ScoutBee*(•) in Line 22 of **Algorithm 1**). This **while** loop is repeated until the termination condition is satisfied. The solution that has the largest fitness value in the final set of solutions F is the final solution.

Algorithm 1 The ABC algorithm

```
1:  Initialize the set of solutions $F = \{x_1, x_2, \cdots x_{|F|}\}$ and set $l_m = 0$, $m = 1, 2, \cdots, |F|$
2:  Evaluate each solution in $F$ to obtain its fitness $f(\bullet)$
3:  while not terminated do
4:    for each solution $x_m$ do
5:      $\widetilde{x}_m \leftarrow EmployedBee(x_m)$
6:      if $f\left(\widetilde{x}_m\right) > f(x_m)$ then $x_m \leftarrow \widetilde{x}_m$, $l_m \leftarrow 0$
7:      else $l_m \leftarrow l_m + 1$
8:      end if
9:      $\widehat{x}_m \leftarrow x_m$
10:   end for
11:   for each onlooker bee do
12:     select a solution $x_m \in F$ with the probability $f(x_m)/\sum_{n=1}^{|F|} f(x_n)$, $\widetilde{x}_m \leftarrow OnlookerBee(x_m)$
13:     if $f\left(\widetilde{x}_m\right) > f\left(\widehat{x}_m\right)$ then
14:       $\widehat{x}_m \leftarrow \widetilde{x}_m$
15:     end if
16:   end for
17:   for each solution $x_m$ do
18:     if $f\left(\widehat{x}_m\right) > f(x_m)$ then $x_m \leftarrow \widehat{x}_m$, $l_m \leftarrow 0$
19:     else $l_m \leftarrow l_m + 1$
20:     end if
21:     if $l_m > limit$ then
22:       $x_m \leftarrow ScoutBee(x_m)$;
23:     end if
24:   end for
25: end while
```

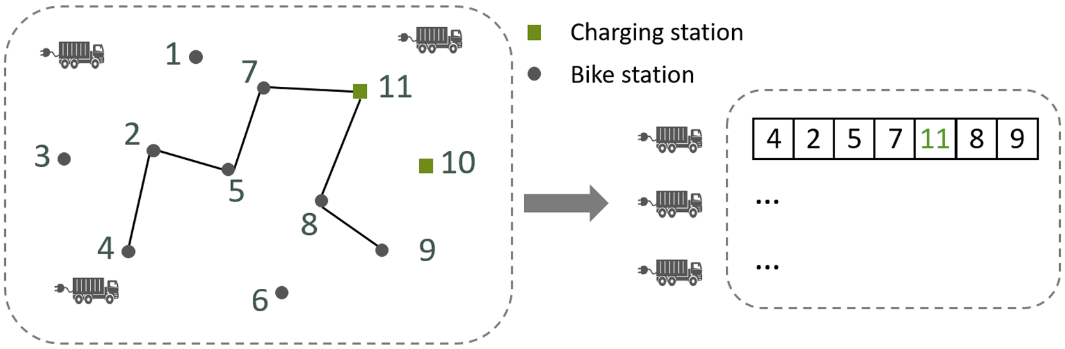


Fig. 4. Encoding of a BEV route.

4.2. Solution representation

As an example, Fig. 4 illustrates the solution representation used in the ABC algorithm. The first BEV is initially located at Station 4 in this planning horizon. This BEV recharges at charging station 11 after successively visiting bike stations 2, 5, and 7. After being fully recharged at charging station 11, this BEV continues onward to visit bike stations 8 and 9. This visiting sequence (route) can be encoded using the node index (right panel in Fig. 4). Once the visiting sequence of each BEV is determined, its loading instructions, arrival times, and SOCs are computed whenever necessary.

4.3. Solution initialization

The initial location and SOC at time w^0 for a given BEV are known at the time of planning, and the initial SOC is always guaranteed to successfully support the BEV in reaching its nearest charging station. To compute an initial BEV route, successive stops are appended one-by-one to the partial route. The feasibility of the next stop is also checked in terms of whether the SOC of a BEV leaving a candidate station can support the BEV to reach its nearest charging station. Specifically, the next stop of the BEV is determined in the following way:

1. If the current SOC of the BEV can support it to visit any bike station and afterward the nearest charging station to this bike station, then a restricted candidate list (RCL) is constructed, based on the BEV travel time from the current stop $s^{'}$: RCL $= \{s | s \in \breve{N} \text{ and } t_{\min} \leqslant t_{s^{'},s} \leqslant t_{\min} + \varepsilon(t_{\max} - t_{\min})\}$, where $t_{\max} = \max\limits_{s \in \breve{N}} t_{s^{'},s}$, $t_{\min} = \min\limits_{s \in \breve{N}} t_{s^{'},s}$, and $\varepsilon \in (0, 1)$. Then, a random bike station s is selected from the RCL, and appended after $s^{'}$. The arrival time ι_s at the appended stop, and the SOC of the BEV, are computed accordingly.
2. Otherwise, the next stop is the nearest charging station to the current stop.

These steps are repeated until there are no unvisited bike stations (i.e., $\breve{N} = \phi$) or the arrival time at the last stop exceeds the planning horizon (i.e., $\iota_s > w^{\mathrm{p}}$).

4.4. Computation of the loading instructions

The loading instructions are computed after a route is generated. According to the definition in Section 3.3.1, a given bike station $s \in \widehat{N}$ is associated with a unique arrival time ι_s. Thus, using an initial inventory $I^s_{w^0}$ at time w^0, the penalty cost for unmet demand during the period $[w^0, \iota_s]$ is estimated as

$$P^1_s = \beta \bullet \mathrm{EB}\left(\mathrm{I}^{\mathrm{s}}_{\mathrm{w}^0}, \mathrm{w}^0, \iota_{\mathrm{s}}\right) + \gamma \bullet \mathrm{EL}\left(\mathrm{I}^{\mathrm{s}}_{\mathrm{w}^0}, \mathrm{w}^0, \iota_{\mathrm{s}}\right) \tag{4.1}$$

and the expected inventory level at time ι_s is

$$EI_s = EI\left(I^s_{w^0}, w^0, \iota_s\right). \tag{4.2}$$

These two values are fixed for any $s \in \widehat{N}$, because all of the inputs are known.

The forecasted penalty cost of s during the period $[\iota_s, w^{\mathrm{f}}]$ depends on the quantities of bikes loaded/unloaded there (lq_s), and is determined as

$$P^2_s(EI_s - lq_s) = \beta \bullet \mathrm{EB}\left(\mathrm{EI}_{\mathrm{s}} - \mathrm{lq}_{\mathrm{s}}, \iota_{\mathrm{s}}, \mathrm{w}^{\mathrm{f}}\right) + \gamma \bullet \mathrm{EL}\left(\mathrm{EI}_{\mathrm{s}} - \mathrm{lq}_{\mathrm{s}}, \iota_{\mathrm{s}}, \mathrm{w}^{\mathrm{f}}\right). \tag{4.3}$$

We determine the loading instruction $LQ_v = \{lq_{r^v_1}, lq_{r^v_2}, \cdots lq_{r^v_{|RT_v|}}\}$ for BEV $v \in V$ such that the total penalty cost incurred at all bike stations in the route is minimized. We follow the dynamic programming algorithm developed by Zhang et al. (2017), to which we add additional considerations regarding the visits to charging stations.

We associate a label ρ^k_a and a preceding load φ^k_a with each stop $r_a \in RT_v$ (for concision, we omit the superscript v of r_a here). ρ^k_a indicates the minimum total penalty cost incurred at all bike stations visited before or at r_a, where the BEV leaves r_a with a load k ($k = 0, 1, \cdots, Q$), and φ^k_a is the corresponding load of the BEV when it arrives at r_a. We assume that the BEV arrives at the first stop r_1 with a load k_0. An integer $\eta \in \{-Q, \cdots, Q\}$ indicates the possible loading (>0) or unloading (<0) quantity at a stop, and ρ^k_a and φ^k_a for the a^{th} stop are determined from $a = 1$ to $a = |RT_v|$, by following these rules:

1. If $a = 1$ and $r_a \in S$, then for each $\eta \in \{-Q, \cdots, Q\}$,

$$\rho^{k_0+\eta}_1 = \begin{cases} P^1_{r_1} + P^2_{r_1}(EI_{r_1} - \eta), & \text{if } 0 \le k_0 + \eta \le Q, 0 \le EI_{r_1} - \eta \le c_{r_1} \\ +\infty, & \text{otherwise} \end{cases} \tag{4.4}$$

and

$$\varphi^k_1 = k_0, \ \ k = 0, 1, \cdots, Q. \tag{4.5}$$

2. If $a = 1$ and $r_a \in C$, then for each $k = 0, 1, \cdots, Q$,

$$\rho_1^k = \begin{cases} 0, & \text{if } k = k_0 \\ +\infty, & \text{otherwise} \end{cases} \tag{4.6}$$

and

$$\varphi_1^k = k_0. \tag{4.7}$$

3. If $a > 1$ and $r_a \in S$, then for each $k = 0, 1, \cdots, Q$, set

$$\eta_a^k = \underset{\eta \in \{-Q, \cdots, Q\}}{\text{argmin}} \left\{ \rho_{a-1}^{k-\eta} + P_{r_a}^1 + P_{r_a}^2 (EI_{r_a} - \eta) | 0 \le k - \eta \le Q, 0 \le EI_{r_a} - \eta \le c_{r_a} \right\}, \tag{4.8}$$

then

$$\rho_a^k = \rho_{a-1}^{k-\eta_a^k} + P_{r_a}^1 + P_{r_a}^2 \left(EI_{r_a} - \eta_a^k \right) \tag{4.9}$$

and

$$\varphi_a^k = k - \eta_a^k. \tag{4.10}$$

4. If $a > 1$ and $r_a \in C$, then for each $k = 0, 1, \cdots, Q$,

$$\rho_a^k = \rho_{a-1}^k \tag{4.11}$$

and

$$\varphi_a^k = k. \tag{4.12}$$

The minimum total penalty cost of all bike stations in this route is determined as $\min\{\rho_{|RT_v|}^k \big| k = 0, 1, \cdots, Q\}$, and $k^* = \underset{k \in \{0, \cdots, Q\}}{\text{argmin}} \rho_{|RT_v|}^k$ is the load of the BEV when it leaves the last stop $r_{|RT_v|}$. The load of the BEV when it arrives at the stop $r_{|RT_v|}$ (i.e., the load of the BEV when it leaves the stop $r_{|RT_v|-1}$) is $\varphi_{|RT_v|}^{k^*}$. Thus, the loading/unloading quantity is calculated as $lq_{r_{|RT_v|}} = k^* - \varphi_{|RT_v|}^{k^*}$. Similarly, the load of the BEV when it arrives at the stop $r_{|RT_v|-1}$ (i.e., the load of the BEV when it leaves the stop $r_{|RT_v|-2}$) is $\varphi_{|RT_v|-1}^{\varphi_{|RT_v|}^{k^*}}$ in the optimal case. The remaining aspects are determined using the same rule, and the loading/unloading quantities at all of the stops are thus obtained.

4.5. *Evaluation of solutions*

After computing the loading instructions of each BEV, the total penalty cost for all of the visited bike stations in $\widehat{N}$ can be calculated. The penalty cost for the unvisited bike stations in $\breve{N}$ is calculated using Equations (3.6) and (3.7). The charging costs are derived from the SOC of each BEV when it arrives at each charging station. Therefore, the objective value is calculated using Equation (3.10), in which P is defined by Equation (3.9). The fitness function $f(\bullet)$ in **Algorithm 1** is defined as the reciprocal of the objective function.

4.6. *Search frameworks for the three types of bees*

To generate new solutions, we use some operators to alter the visiting sequence of bike and charging stations. After that, the loading/unloading instructions are computed.

Operators: There are five operators in the neighborhood search: three for bike stations (set denoted by *NS*) and two for charging stations (set denoted by *NF*).

The three operators for bike stations comprise the exchange of two bike stations in separate routes, the reverse of a section in a single route, and the removal of a random bike station. After the implementation of a bike station operator, the time and the SOC of a BEV upon its arrival at each stop are computed. The *feasibility* of each stop is checked, with respect to three criteria: (1) the arrival time must not exceed the planning horizon, (2) the SOC must take a value that ranges from 0 to the battery capacity, and (3) the SOC must be sufficient to ensure that the BEV can reach its nearest charging station. The stops subsequent to the first stop that violates any of these criteria are deleted from the route. If all of the stops in a route satisfy the above three criteria, one of any unallocated bike stations is randomly selected and appended to the end of the route, and the same feasibility check is applied. If the SOC of the BEV is infeasible, the nearest charging station is appended to the end of the route. This appending of nodes is repeated until the makespan covers the planning horizon, or the set of unvisited bike stations $\breve{N}$ is empty.

The two operators for charging stations (i.e., the set *NF*) comprise the replacement of a visited charging station and the relocation of a charging station in a route. The best improvement strategy is adopted for the neighborhood search by using the operators in the set *NF*. That is, for the replacement operator, the charging station that leads to the minimum objective value of all unvisited charging stations is selected as the replacement for the visited charging station; and for the relocation operator, the visited charging station is inserted into a position in the visiting sequence that leads to the minimum objective value. The same feasibility check as mentioned above is applied after the implementation of a charging station operator.

EmployedBee (•) and *OnlookerBee* (•): The employed bees and onlooker bees share the same neighborhood-search framework, as summarized in **Algorithm 2**. The main difference between these two types of bees lies in their selection of food sources, as explained in **Algorithm 1**. Neighborhood operators for bike stations are applied to a solution x in random order. After selecting a random neighbor solution $\widetilde{x} \in NS_h(x)$ (Line 2 in **Algorithm 2**), a random charging-station operator is applied to solution $\widetilde{x}$ to find the best neighbor solution $\widehat{x}$ (Line 4 in **Algorithm 2**). If solution $\widehat{x}$ has a larger fitness value than solution x, x is replaced with $\widehat{x}$, and the same neighborhood operator is repeated (Line 7 in **Algorithm 2**) until solution x cannot be improved. Otherwise, the next bike-station operator is applied (Line 10 in **Algorithm 2**).

```
Algorithm 2 The subroutines EmployedBee (•) and OnlookerBee (•)
Input: solution $x$ with a fitness value of $f(x)$
1:   Randomly number the neighborhood operators for bike stations, and set $h = 1$
2:   Randomly select $\widetilde{x} \in NS_h(x)$, where $NS_h(\bullet)$ is the set of neighbor solutions obtained by bike station operator $h$
3:   if there are charging stations in $\widetilde{x}$, then
4:     Randomly select a charging station operator $d$, where $d = 1$ or 2, and select the best neighbor solution $\widehat{x} \in NF_d(\widetilde{x})$, where $NF_d(\bullet)$ is the set of neighbor
     solutions obtained by charging-station operator $d$
5:   end if
6:   if $f(\widehat{x}) > f(x)$ then
7:       $x \leftarrow \widehat{x}$, go to Line 2
8:   end if
9:   If $h < 3$then
10:      $h \leftarrow h + 1$, go to Line 2
11:  end if
```

ScoutBee (•): A scout bee replaces input solution x with its neighbor solution, instead of using a randomly generated solution (as in the original ABC algorithm) (Szeto et al., 2011). This neighbor solution is generated by randomly removing a bike station from every BEV route in x. The feasibility check and node-appending process (appending of bike and charging stations) introduced above for neighborhood operators are also applied in this context. The neighbor solution replaces solution x, regardless of whether the fitness value is improved.

The evaluation process is applied whenever a new visiting sequence is generated. To avoid recalculations, ρ_a^k and φ_a^k for the stops before the revised section of the route can be retained.

5. Computational experiments

We used system data (https://www.capitalbikeshare.com/system-data) from Capital Bikeshare (Washington, DC) in the computational experiments. The proposed method was coded in Visual Studio 2013 using C++ and executed on a high-performance computer equipped with two 10-core Intel Xeon E5-2600 v3 (Haswell) processors and 96 GB of physical memory.

We use a 24-hour clock in this section. Section 5.1 describes the dataset used in the experiments and the values of parameters. Section 5.2 discusses the effects of horizon lengths on the objective value. Section 5.3 demonstrates the effects of the ratio of penalties β/γ on the repositioning performance. Section 5.4 illustrates three aspects of the effects of charging-related settings: battery capacities VB, the number of charging stations $|C|$, and multiple charging technologies. Section 5.5 compares the results from the canonical GA with those from the proposed ABC algorithm. Section 5.6 shows the runtime of the rolling horizon framework.

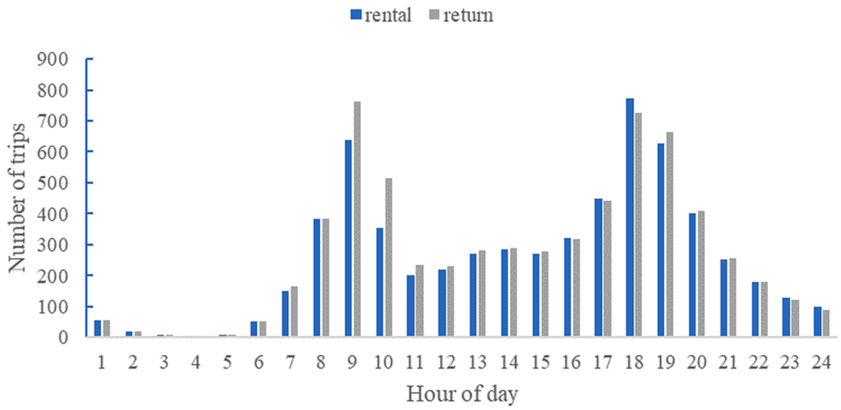


Fig. 5. Total average rentals and returns at all target bike stations in each hour of a day.

5.1. Data description and parameter settings

We focused on a subset of bike stations from Capital Bikeshare (Washington, DC) that comprised the 100 most active bike stations with capacities varying from 15 to 55. The data were preprocessed according to the format presented by Vogel et al. (2011). We extracted all of the trips on working days (Monday to Friday) in October 2019. The average number of rental and return events at each of these stations in each hour over a day were used to generate the rental and return rates at each of these stations on an hourly basis. This afforded 48 attributes that describe the daily demand for bikes and lockers for each bike station. Fig. 5 shows the total daily average rentals and returns at the 100 stations during each hour of a day. As evident in the figure, there were generally two peaks in daily demand for both rentals and returns: at 08:00–09:00 and at 17:00–18:00. There is a small peak in the demand at 13:00–14:00.

Some bike stations had similar patterns of rentals and returns. For example, Fig. 6 shows the average rentals and returns in each hour of a day at a bike station that had a similar demand for bikes (blue line) and for lockers (grey line). Some other bike stations had a single peak in demand for both rentals and returns, and these two peaks did not overlap. For example, the bike station depicted in Fig. 7 had a single peak of rentals in the morning and a single peak of returns in the afternoon. Similarly, the bike station shown in Fig. 8 had a single peak of rentals in the afternoon and a single peak of returns in the morning.

There were an average of 10,246 trips in the entire area (i.e., including all of the bike stations in use) per day. We followed the instance-generation steps of Brinkmann et al. (2019) and generated a realization by randomly drawing 10,246 trips with replacement from all of the trips during workdays in October 2019. The initial inventory at each bike station was randomly generated as a value from 0 to the station capacity. The simulated BSS evolved with all of the generated rental/return events and repositioning activities, as introduced in Section 3.4. When a vehicle arrived at a bike station, it loaded/unloaded a greedy number of bikes that was closest in value to the planned quantity. The Euclidean distances between nodes were used. Three BEVs were used in the experiments, and the vehicle speed was assumed to be 25 km/h.

To estimate the forecasted inventory and unmet demand at each bike station, the discretization level in the approximation of the transition probability matrix was set as 15 min (i.e., $\delta = 15$ min), and the value of M in Equation (3.3) was set as 300. The values of parameters in the algorithm were tuned with the planning horizon set to 4 h and the forecasting horizon set to 6 h. The value of the weight α in the objective function was assumed to be 0, unless otherwise specified (the effects of α are investigated in Section 5.4.3). We first tuned the number of food sources (or solutions, i.e., $|F|$), which was equal to the number of employed bees (and the number of onlooker bees), in a sufficient runtime. The threshold for the scout-bee phase was 10 times the network size (i.e., 1000 for this network). The results suggested that the setting of 10 food sources gave the best convergent solution within 15 min (the discretization level mentioned above). Then, the value of *limit* was tuned; we applied the multipliers of 1, 5, 10, and 15 to the network size, which revealed that five times the network size gave the best solution. Therefore, the parameters for the algorithm were set as follows: the number of food sources was 10, the threshold for the scout bee phase was five times the network size, and the computational time for calculation (i.e., t_c) was 15 min.

5.2. Effects of different lengths of planning horizon

We followed the settings of Usama et al. (2019), and thus the battery capacity was assumed to be 16 kWh, and the energy consumption rate during repositioning was approximated as 8 kWh per h (i.e., the battery autonomy was 2 h). The charging facilities were assumed to be identical (i.e., possess the same charging technology) at all 10 charging stations (a charging station may have many charging facilities), which were randomly distributed between the bike stations. Under these conditions, a full-cycle charge was assumed to take half an hour. The penalties for failed rentals or returns were assumed to be identical.

Three different instances, s1, s2, and s3, were generated, with different initial inventory levels at bike stations, different realizations

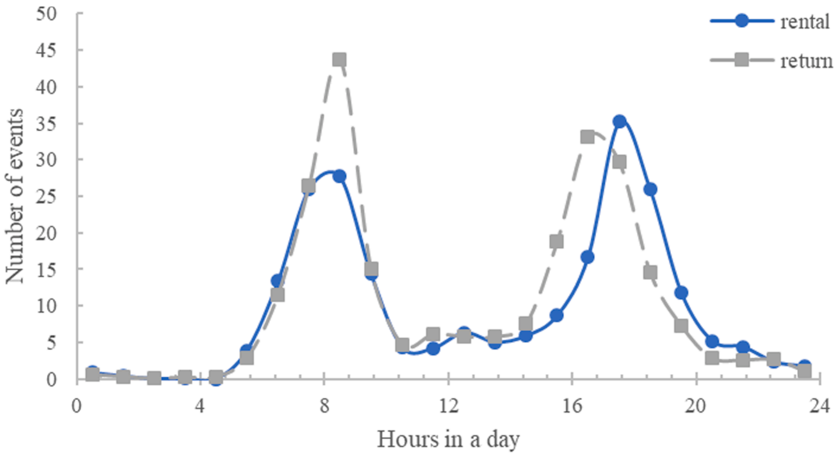


Fig. 6. An illustrative bike station with two peaks for both rentals and returns.

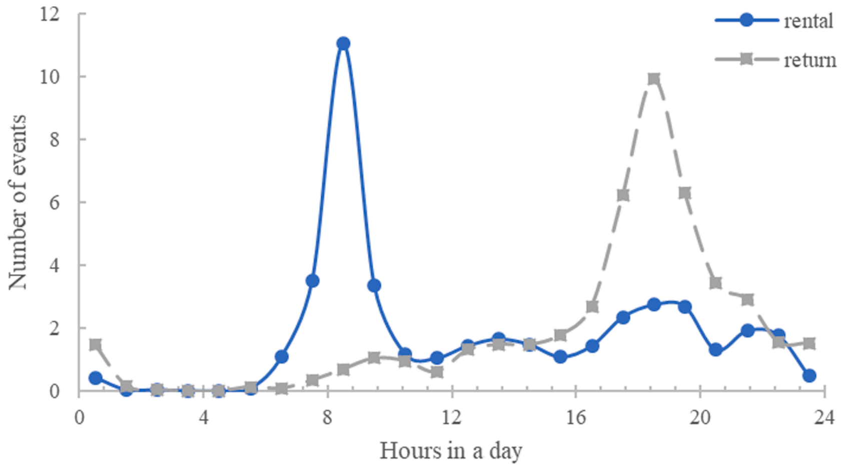


Fig. 7. An illustrative bike station with a single peak for rentals in the morning and a single peak for returns in the afternoon.

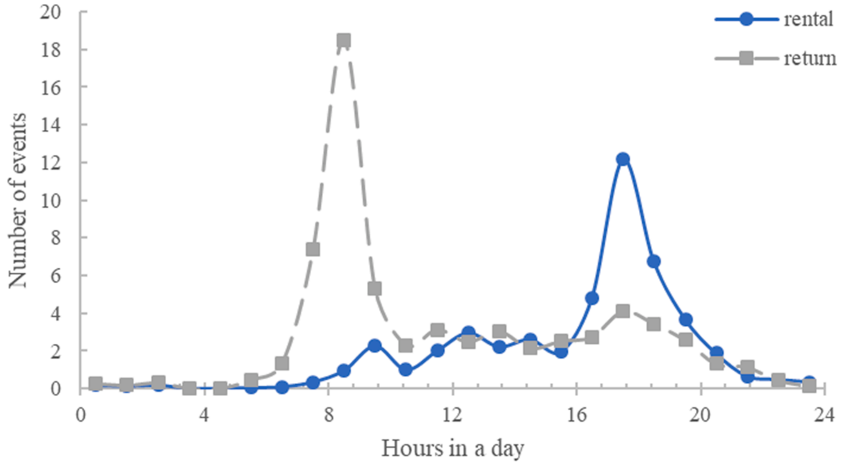


Fig. 8. An illustrative bike station with a single peak for returns in the morning and a single peak for rentals in the afternoon.

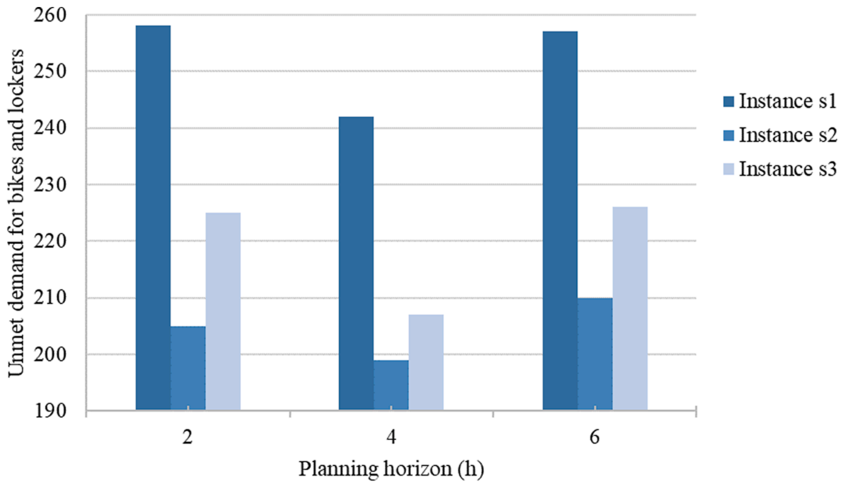


Fig. 9. Realized unmet demand for bikes and lockers for various planning-horizon lengths (forecasting horizon = 6 h).

of user demand, and different initial statuses of BEVs (i.e., in terms of their location, load of bikes, and SOC). The rolling step t_r was set as 2 h, and we focused on the realized number of failed rentals (due to a shortage of bikes) and returns (due to a surplus of bikes) from 08:00 to 10:00 (i.e., the simulation period).

We maintained a fixed forecasting-horizon length (6 h), and varied the planning-horizon length (from 2 to 4 to 6 h) to investigate its effect on the repositioning performance. Fig. 9 shows the total unmet demand (which includes failed rentals and returns) for different planning-horizon lengths for the three instances. It can be seen that a planning horizon of 4 h generates less unmet demand than a planning horizon of 2 h, as the latter equals the rolling step t_r. This suggests that considering the possible stops beyond the rolling step may help to improve the repositioning performance. However, a planning horizon of 6 h generates greater unmet demand than a planning horizon of 4 h. This may be because the route and loading instructions are optimal for the bike stations visited within the planning horizon, but are not necessarily optimal for the bike stations visited within the rolling step (i.e., the period during which the planned activities are actually carried out), which is shorter in duration than the planning horizon.

5.3. *Effects of penalties for failed rentals and returns*

We assigned different ratios of penalties to each failed rental and return to investigate its effect on the realized unmet demand. The repositioning task was planned with β/γ set to 10:1 and 1:10. We rolled the forecasting horizon forward every 2 h, and recorded the realized number of failed rentals (due to a shortage of bikes) and returns (due to a surplus of bikes) from 8:00 to 12:00. The results are shown in Table 4. In Instance s1, if a higher penalty was assigned to failed rentals (i.e., β/γ is 10:1), the number of failed rentals and returns were 58 and 247, respectively. In contrast, if a higher penalty was assigned to failed returns (i.e., β/γ is 1:10), the number of failed rentals increased to 89, whereas the number of failed returns decreased to 234. Instances s2 and s3 followed the same pattern. This shows that the realized failures of a certain type of demand (for either bikes or lockers) could be reduced by assigning a higher penalty to these failures.

However, irrespective of how the ratio was changed, there were more failed returns than failed rentals. This may be because the demand for returning bikes was greater than that for renting bikes during the target period such that the system had a surplus of bikes. This situation is evident in Fig. 5: from 08:00 to 11:00 (corresponding to the abscissa values of 9, 10, and 11), more bikes are returned than rented. However, the repositioning ability of BEVs (in terms of the size of the BEV fleet and BEV capacity) was limited; thus, surplus bikes may not have been transported away in time.

5.4. *Effects of charging-related settings*

In this section, we describe our three-part investigation of the effects of charging-related settings on the DBRP. First, we used BEVs with different battery capacities to plan the repositioning task and compared the unmet demand and the distribution of the charging time. Second, we varied the number of charging stations in the network and compared the resultant unmet demand. Third, we set different charging technologies in the charging stations and compared the unmet demand and charging costs of the resulting scenarios.

5.4.1. *Effect of different battery capacities*

We assumed a fixed charging speed and increased the battery capacity of each BEV from 16 kWh (autonomy = 2 h) to 32 kWh (autonomy = 4 h) and 48 kWh (autonomy = 6 h). Therefore, the full-cycle charging time for a battery with 4-h autonomy increased to 1 h, and that for a battery with 6-h autonomy increased to 1.5 h. All of the vehicles were regarded as fully charged at 08:00, and the penalties for failed rentals and returns were identical. We focused on the realized unmet demand incurred from 08:00 to 24:00, during which the average charging time of each vehicle was almost identical for batteries of different capacities (i.e., 185 min, 170 min, and 178 min). The realized unmet demand with various battery capacities during each rolling step is shown in Fig. 10. The distribution of charging time for three BEVs with different battery capacities is presented in Fig. 11.

It can be found from Fig. 10 that a larger battery capacity led to a lower total unmet demand over the simulated 16-h period. Specifically, from 08:00 to 10:00, the 2-h autonomy scenario generated the most unmet demand of the three capacity settings. This is because the 2-h autonomy period was equal to the rolling step, during which a BEV must recharge to avoid running out of battery power. Given that this required a detour to a charging station, which takes extra time, a charging action required more time than that spent at the charging station. In addition, the system experienced high usage from 08:00 to 10:00, which increased the unmet demand.

In the subsequent three rolling steps (i.e., steps during [10:00, 16:00]), the differences in the unmet demand between the three capacity settings decreased. Then, in the step [16:00, 18:00], the 6-h autonomy scenario generated the most unmet demand (79 vs. 68 and 60). This may be because the unmet demand during step [16:00,18:00] was somewhat controlled by repositioning during the step

Table 4
Realized unmet demand with different penalties for failed rentals and returns.

Instance	β/γ = 10:1		β/γ = 1:10	
	Failed rentals	Failed returns	Failed rentals	Failed returns
s1	58	247	89	234
s2	62	180	81	146
s3	79	287	85	272

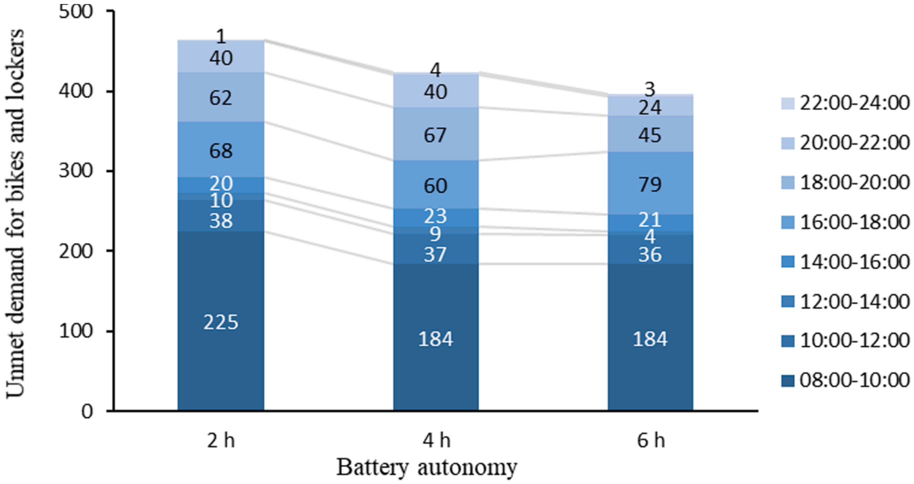


Fig. 10. Realized unmet demand for bikes and lockers from 08:00 to 24:00, resulting from the use of battery electric vehicles with various battery capacities.

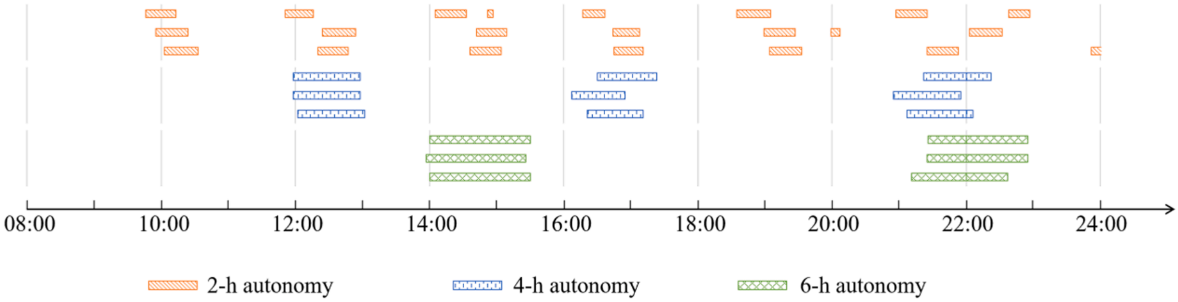


Fig. 11. Distribution of the charging time from 08:00 to 24:00 h for battery electric vehicles with various battery capacities.

[14:00,16:00], during which the vehicles were mainly charging. In addition, the system experienced high usage during the step [16:00,18:00] and thus its performance was highly reliant on advanced repositioning during the step [14:00,16:00]. This effect can also be seen in step [18:00, 20:00], during which the 4-h autonomy scenario generated the most unmet demand because the vehicles were charging during the previous step [16:00,18:00].

The overall trend of the reduction in the total unmet demand with increasing battery capacities may be because a BEV with a larger battery capacity detoured the least to charging stations, which minimized its extra distance traveled and time expended. This indicates that the full charging of BEVs should be scheduled to occur during non-busy periods.

5.4.2. *Effect of the number of charging stations*

We re-considered the three instances in Table 4. For each instance, the number of charging stations was increased from 5 to 25, with a step of 5. For each number of charging stations examined, 10 different configurations of their locations were randomly constructed. We focused on the planning horizon from 08:00 to 11:00 to analyze the effects of the number of charging stations (i.e., $|C|$) on unmet demand.

Fig. 12 shows the boxplots of the unmet demand for various values of $|C|$. Increasing $|C|$ from 5 to 10 reduced the unmet demand in all three instances, as the time spent on detouring to charging stations decreased. However, further increasing the number to 15, 20, and 25 did not significantly reduce the unmet demand, as the time spent on detouring to charging stations cannot be further reduced without changing or optimizing the locations of these stations.

5.4.3. *Effect of multiple charging technologies*

In the previously described experiments, we assumed that there were identical charging technologies at all charging stations (locations). In this experiment, we examined scenarios in which there were combinations of different charging technologies at the charging stations. Felipe et al. (2014) categorized the three main types of charging technologies as follows: slow (S), fast (F), and very fast (VF). These technologies have different charging speeds, costs, and availabilities, as listed in Table 5. A fixed cost was attached to each charge, and it was assumed that S charging technology could only be used from 00:00 to 06:00. Therefore, we performed separate experiments for daytime and nighttime repositioning.

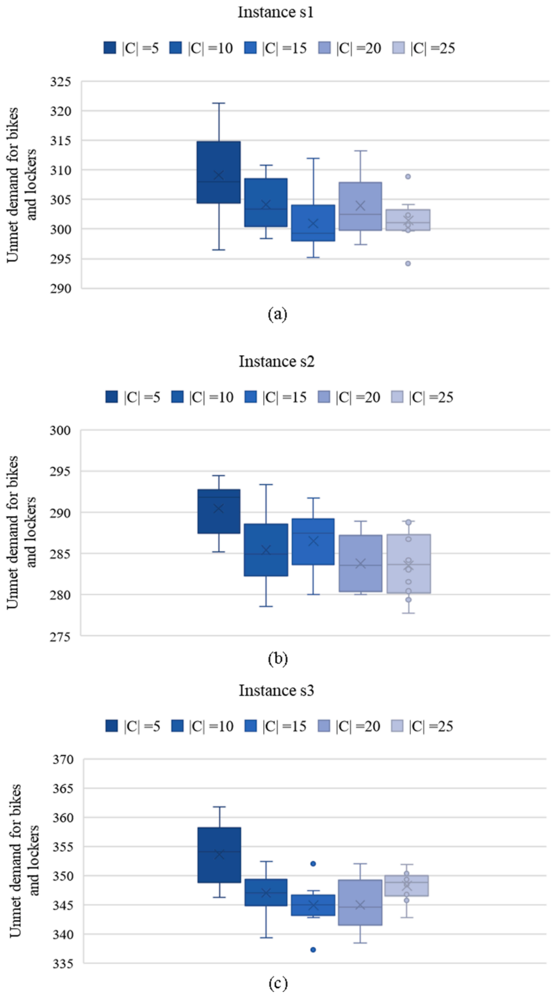


Fig. 12. Boxplots of the unmet demand for bikes and lockers with different numbers of charging stations: (a) Instance s1, (b) Instance s2, and (c) Instance s3.

There were three charging scenarios for *daytime repositioning*:

F charging only: 10 charging locations, all equipped only with F charging technology

VF charging only: 10 charging locations, all equipped only with VF charging technology

Multitechnology (M) charging: 10 charging locations, all equipped with F and VF charging technologies

The penalties for each failed rental (β) and return (γ) were both \$1. The value of α varied from 0.01 to 100 for a tenfold step. The planning horizon was set as [08:00, 11:00] and the program was run 10 times for each value of α. Fig. 13 shows the average penalty costs with various values of α for unmet demand and average charging costs in the above three scenarios. Table 6 presents the p-values of the two-tailed t-test of penalty costs and charging costs of M vs. F/VF charging scenarios.

Fig. 13 shows that in all three scenarios, as the value of α increased, the penalty cost for unmet demand increased and the charging cost decreased, as expected. At certain values of α (when $\alpha \in \{0.01,0.1,1,10\}$), the penalty cost for unmet demand with only F charging

Table 5
Available charging technologies.

	Charging technology		
	Slow	Fast	Very fast
Power rating g_i	0.06 kWh/min	0.33 kWh/min	0.75 kWh/min
Unit charging cost cr_i	\$0.01/min	\$0.067/min	\$0.16/min
Full-cycle charging time	4.5 h	48 min	21 min
Availability	Home charging	Public charging station	Public charging station
Fixed cost cr_0		\$2.53	

technology available was greater than that with only VF counterparts. When α equaled 100, the difference between the penalty costs of F and VF charging scenarios was not statistically significant. The charging cost with only F charging technology available was less than that with only VF counterparts for all values of α, because there was a greater charging cost per kilowatt-hour for VF charging.

In the M charging scenario (in which both charging technologies were available), when $\alpha \in \{0.01, 0.1, 1\}$, the penalty costs and charging costs were closer to the scenario with only VF charging, which is supported by the p-values in Table 6. When $\alpha \in \{10, 100\}$, the penalties and charging costs of the M charging scenario were closer to the scenario with only F charging. This might be because when both technologies were available and the weight of the charging cost was small (e.g., 0.01, 0.1, or 1), the operator sacrificed the charging costs to reduce the penalty costs for unmet demand. The charging cost of the M charging scenario was no larger than that of the VF charging scenario. In contrast, the penalty cost of the M charging scenario was no less than that of the VF charging scenario. When the weight of the charging cost was 10 or 100, the charging cost was more important than the penalty cost for unmet demand. Therefore, the operator decided to save charging costs. The differences between the penalty costs of three scenarios were not statistically significant for $\alpha = 10$ and 100. In summary, the performance of using multiple technologies varies between the performance of two single technologies with different values of α.

The average charging time in the planning horizon with different charging technologies is presented in Fig. 14. Using VF charging technology took less time than using F charging technology. The charging time for the M charging scenario was relatively stable with the variation in α. However, in the single technology (F or VF charging) scenarios, the charging time decreased as the value of α increased. This trend is consistent with the trend of charging costs depicted in Fig. 13, as a shorter charging time resulted in lower charging costs.

We compared two scenarios for *nighttime repositioning* to investigate the effect of using S charging at night:

Use of public charging facilities: BEVs used public charging facilities (at which F and VF charging technologies are available) from 00:00 to 06:00.

Use of depot charging: BEVs park at depots and used S (home) charging from 00:00 to 06:00.

As the full-cycle charging time for an S charge is 4.5 h, the vehicles could be fully recharged before 06:00. We ignored the effect of the location of S charging facilities on the repositioning planning. Thus, we assumed that there were sufficient no-service hours for BEVs to return to any depot, fully recharge, and return to work.

To illustrate the effect of vehicle activities during nighttime on the future unmet demand, a simulation period from 00:00 to 08:00 was used (i.e., one extra rolling step was added after the target period [00:00, 06:00]). Public charging facilities were available in both scenarios for the period from 06:00 to 08:00. The resultant penalty costs and charging costs were grouped into two time-windows: [00:00, 06:00] and [06:00, 08:00]. Fig. 15 shows the results for two values of α in three instances ($\alpha = 0.01$ for the left column and $\alpha = 10$ for the right column).

In all six instances, the penalty costs for unmet demand when using public charging facilities were less than those for unmet demand when using home charging. In contrast, the costs for charging at public facilities were greater than those for charging at home in these instances. This suggests that performing nighttime repositioning using public charging facilities could better relieve the unmet demand incurred in the daytime than using home charging, albeit at higher charging costs. The penalty costs were mainly due to the unmet demand during time window [06:00, 08:00], when the system began to experience higher usage (see Fig. 5). Therefore, the penalty costs in the two scenarios during time window [06:00, 08:00] were larger than those during time window [00:00, 06:00]. In addition, in each of the six instances, the penalty cost of using home charging was greater than the penalty cost of using public charging during time window [00:00, 06:00]. This reveals that performing nighttime repositioning when using public charging facilities could reduce "instant" unmet demand. Although public charging facilities were available in the two scenarios during time window [06:00,

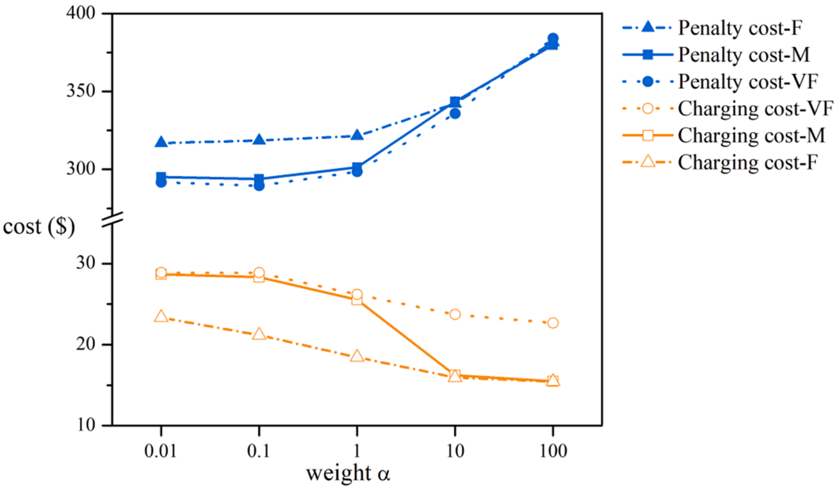


Fig. 13. Average penalty costs and charging costs with different charging scenarios (multitechnology (M), fast (F) charging, and very fast (VF) charging).

Table 6
p-values of the *t*-test of penalty costs and charging costs for multi-technology (M) vs. fast (F)/very fast (VF) charging scenarios.

α		0.01	0.1	1	10	100
Penalty cost	M-F	**2.6E-09**	**7.5E-10**	**2.1E-05**	0.7746	0.7267
	M-VF	0.1818	0.0945	0.4058	0.0582	0.4978
Charging cost	M-F	**5.2E-06**	**1.4E-06**	**5.8E-06**	0.1531	0.1974
	M-VF	0.6733	0.0194	0.4794	**2.2E-17**	**9.4E-15**

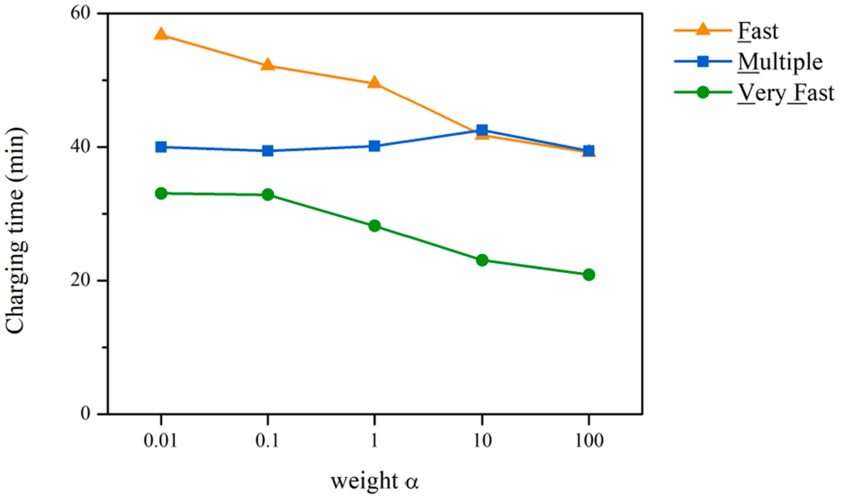


Fig. 14. Charging time for step [08:00,11:00] with different charging technologies.

08:00], at this time, the penalty cost of using public charging was smaller than that of using home charging in each of the six instances. This suggests that performing nighttime repositioning when using public charging facilities could also reduce “future” unmet demand. The BEVs were recharged more than once, and they used the more expensive charging technologies during time window [00:00, 06:00] when using public charging facilities (i.e., F and VF charging) than when using home charging (i.e., S charging); therefore, the former scenario had the highest recharging costs.

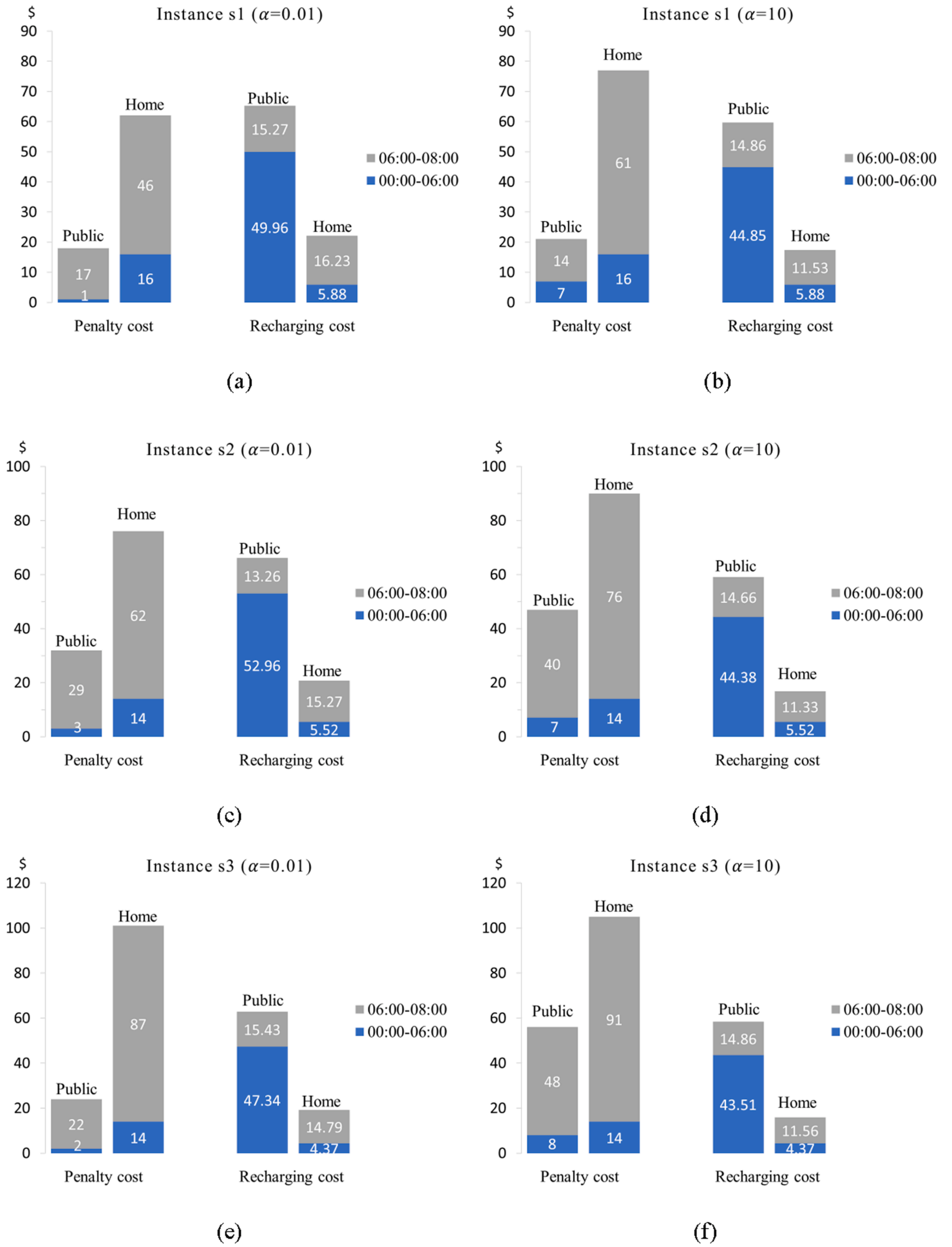
5.5. Comparison of the canonical GA and the ABC algorithm for solving subproblems

We compared the ability of the canonical GA and the proposed ABC algorithm to solve the subproblem defined in Section 3.3. The GA is a population-based metaheuristic that mimics the evolution of creatures in nature. It has been widely used to generate solutions to well-known problems, such as the traveling salesman problem (e.g., Braun, 1990) and the pickup and delivery problem (e.g., Hosny and Mumford, 2009). We selected the canonical GA as a benchmark methodology in this study to demonstrate the effectiveness of the proposed ABC algorithm. The two operators used in the GA for generating new solutions comprise the crossover operator, which exchanges the complete route of a random vehicle in two solutions, and the mutation operator, which is the same as the operator in the scout bee phase of the ABC algorithm described in Section 4.6. The crossover rate and mutation rate were determined by tuning and were 0.3 and 0.3, respectively.

We selected 24 different forecasting horizons (i.e., subproblems) and applied the GA and the ABC algorithms separately to generate solutions. Both algorithms were run 10 times on each subproblem, and the final outputs were recorded. The average objective value and the standard deviation of solutions in 10 runs were calculated. To demonstrate the performance of the proposed ABC algorithm, we computed the percentage improvement in the average objective value of the ABC algorithm relative to the GA, and performed a *t*-test; these results are presented in Table 7. On average, the ABC algorithm outperformed the GA by 23.21%, and all of the *p*-values were much smaller than 0.01. In summary, the proposed ABC algorithm exhibited better objective values than the GA, and this difference in performance was statistically significant.

5.6. Runtime of the rolling horizon framework

A given simulation period, e.g., 24 h, had a total runtime that was composed of two components: the optimization time, for computing the routing and loading decisions during each planning horizon, and the simulation time, for deriving the realized repositioning actions and the evolution of the status of repositioning BEVs and bike stations during each rolling step (see Fig. 2 for reference). The optimization for each rolling step was fixed and controlled by the parameter t_c, which is defined in Section 3.2. The



(a)

(b)

(c)

(d)

(e)

(f)

Fig. 15. Penalty costs and recharging costs for using home and public charging facilities.

Table 7
Results from use of the canonical genetic algorithm (GA) and the proposed artificial bee colony (ABC) algorithm to solve subproblems.

Subproblem	Mean		SD		Improvement	*p*-value
	GA	ABC	GA	ABC		
1	355.5	298.2	8.51	3.22	16.13%	7.29E−11
2	57.6	42.9	4.05	0.67	25.52%	6.28E−07
3	69.4	55.4	1.10	0.79	20.10%	2.38E−16
4	133.7	102.0	5.91	2.03	23.71%	2.79E−09
5	255.8	198.6	13.82	4.77	22.37%	4.23E−08
6	146.7	122.3	4.44	0.79	16.65%	4.85E−09
7	40.9	30.4	1.98	0.48	25.62%	7.87E−09
8	20.6	14.0	2.73	0.37	31.90%	1.8E−05
9	339.3	286.0	8.57	3.36	15.72%	1.94E−10
10	53.0	36.6	3.21	0.89	30.97%	1.22E−08
11	61.4	44.4	2.22	1.02	27.68%	5.77E−12
12	144.0	107.8	10.59	3.76	25.18%	3.02E−07
13	278.7	216.2	7.58	2.98	22.41%	7.28E−12
14	146.8	121.5	3.98	1.05	17.22%	1.42E−09
15	37.2	27.6	1.51	0.37	25.75%	1.37E−09
16	15.6	10.3	1.31	0.27	33.75%	9.96E−08
17	402.4	348.5	13.06	7.10	13.39%	8.44E−09
18	45.7	34.0	2.52	0.61	25.66%	2.76E−08
19	46.2	36.2	1.94	0.56	21.75%	1.09E−08
20	98.8	73.2	7.07	0.84	25.89%	6.06E−07
21	210.2	168.9	8.92	3.62	19.68%	6.03E−09
22	132.7	108.5	4.98	1.36	18.27%	1.92E−08
23	29.2	23.0	1.88	0.31	21.24%	1.41E−06
24	8.0	5.5	0.72	0.28	30.50%	1.94E−07

SD = standard deviation.

value of t_c was set as 15 min for all of the experiments in this study (see Section 5.1), because the discretization level for generating the arrival rates of customers was 15 min, and because a longer optimization time may reduce the accuracy of forecasts. Therefore, the total optimization time was solely determined by the number of rolling steps in the simulation period. For example, with a simulation period of 24 h and a rolling step of 2 h, the total optimization time should be 12 rolling steps × 15 min. Note that the optimization time for each rolling step is constant and unaffected by other factors, such as the length of the planning or forecasting horizon. As the simulation of the user activities and repositioning activities was carried out offline using known sequences of rental/return events, the realized status of the system was derived within a very short computational time (less than 0.001 s), which was negligible compared with the optimization time. Thus, the runtime of the rolling horizon framework only related to the number of rolling steps in the examined simulation period.

6. Conclusion and discussion

This study examined a DBRP with BEV and multiple charging technologies. The solution includes stipulated BEV routes, the loading/unloading quantities of BEVs at each visited bike station, and the arrival times of BEVs at visited bike and charging stations. The objective was to minimize the weighted sum of the penalty costs for unmet user demand for bikes and lockers, and the charging costs of repositioning vehicles. Dynamic decisions were made under a rolling horizon framework to incorporate the latest information on bike stations and repositioning BEVs. The ABC algorithm was used to solve each subproblem of the proposed DBRP–BEV, with an embedded dynamic programming algorithm used to compute the loading instructions. Computational experiments were carried out using data from the Capital Bikeshare, a BSS in Washington, D.C., US, to illustrate problem properties and the runtime of the solution method. Experiments were conducted on different aspects of the problem, and the results provide guidance on the practical application of BEVs in the DBRP.

6.1. Summary of main findings

First, in terms of dynamic planning settings, considering a planning horizon that is longer than the planning interval may help to improve the repositioning performance. However, an overlong planning horizon may have adverse effects.

Second, in terms of the penalty cost for unmet demand, the failures of a certain type of demand (for either bikes or lockers) can be reduced by assigning a higher penalty to this failure during planning. Thus, a system operator should adjust the ratio of the penalties for failed rentals and returns according to the operational priorities of a BSS.

Third, in terms of charging-related settings, the use of a BEV with a larger battery capacity resulted in more reduction in total unmet demand. Increasing the number of charging stations to a certain number also reduced the unmet demand, but there was no further improvement after a certain number was reached. In addition, when the weight α of the charging cost in the objective function was increased, the penalty cost for unmet demand increased and the charging cost decreased; overall, as α varied, the penalty cost and charging cost for the M (i.e., F and VF) charging scenario varied between those of the F charging scenario and the VF charging scenario. Therefore, an operator should adjust the weight α to balance the penalty cost and charging cost, and the charging technology can be selected according to this weight (if multiple charging technologies are available). However, at a fixed weight α, the use of M charging technology was not superior to the use of a single charging technology in terms of charging and penalty costs. For nighttime repositioning, the penalty cost for unmet demand when using public charging facilities (F or VF charging scenario) was less than that for using depot charging (i.e., S charging scenario); in contrast, the charging cost for public charging facilities was greater than that for depot charging facilities. This suggests that when performing nighttime repositioning, using public charging facilities can better relieve unmet daytime demand than using home charging, although at higher charging costs.

Four, to the best of our knowledge this is the first study of the use of BEVs in the DBRP. It reveals that public charging facilities and depot charging facilities can be used in this context, in contrast to previous studies that only considered the use of depot charging facilities.

Finally, the proposed solution methodology affords solutions to the DBRP–BEV for a real-world BSS within a reasonable computation time.

6.2. Future research directions

Future research directions could involve the relaxation of some constraints in the proposed DBRP–BEV or the inclusion of more realistic settings, such as those used in the studies of the EVRP. The following are examples of such possible directions. First, the full charging constraint could be relaxed, as partial recharging provides more flexibility to BEVs in terms of when they can cease charging. This relaxation would require additional decisions to be made on the charging amount on each charging occasion, and thus create a more complicated problem than that considered in this study. Second, given the limitation in charging resources, BEVs may not commence recharging as soon as they arrive at a charging station; thus, the queuing time at a charging station should be taken into consideration. This would introduce a complication in making charge-scheduling decisions. Third, although BEVs are a green transport mode, an ICEV has easier access to refueling stations and a longer cruising range than a BEV. In addition, the transition from ICEVs to BEVs may not be completed in one step. It will therefore be important to consider a mixed fleet of ICEVs and BEVs, and to examine the impact of the composition of such a fleet on repositioning performance and on the environment. In future, we will address this by extending the work of Jia et al. (2021), who were the first to consider a mixed fleet of ICEVs and BEVs in the SBRP, to the DBRP.

In addition to these extensions of the DBRP–BEV, the solution algorithm for this type of problem could be further explored. As the newly proposed DBRP–BEV shares some characteristics with the well-addressed BRP and EVRP, algorithms that perform well for the BRP and EVRP could be adapted to solve the DBRP–BEV.

6.3. Implications of the proposed methodology

This study provides a rolling horizon framework to incorporate the latest information in a BSS to enable periodic updating of repositioning decisions. This planning framework is a compromise between computational efforts and predictive accuracy, which makes it suitable for dynamic (or online) decision-making processes. If the computation is fast or the planning interval is small, this framework is close to a real-time planning system. Modern information and communication technology have accelerated the development of mobile application-supported on-demand services in urban mobility. The planning framework in this study could also be applied to other demand-responsive transportation that requires the continual re-optimization of planning in response to new demand, such as ride-sharing.

Similar relocation problems have been examined in other shared mobility scenarios, such as electric (e-) scooter sharing (e.g., McKenzie, 2019) and e-car sharing (e.g., Brendel et al., 2018). E-scooters, like shared bikes, must be periodically relocated, but unlike bikes, they also periodically need recharging. Our proposed solution algorithm could be modified to adapt to new problem settings, such as those for e-scooter relocation and recharging. The relocation of shared cars differs from that of shared bikes and e-scooters, as relocation staff can be only assigned to one vehicle at a time; thus, shared car relocation can be viewed as a full truckload pickup and delivery problem (Nourinejad et al., 2015). Our proposed solution method could also be applied to the routing problem (e.g., Levin, 2017) and vehicle repositioning problem (e.g., Narayanan et al., 2020) of shared autonomous-vehicle services.

CRediT authorship contribution statement

Yue Wang: Conceptualization, Methodology, Investigation, Visualization, Writing - original draft, Writing - review & editing. **W.Y. Szeto:** Conceptualization, Methodology, Writing - original draft, Writing - review & editing, Supervision, Funding acquisition, Project administration.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research was jointly supported by grants from the National Natural Science Foundation of China (No. 71771194), the Guangdong-Hong Kong-Macau Joint Laboratory Program of the 2020 Guangdong New Innovative Strategic Research Fund, the Guangdong Science and Technology Department (2020B1212030009), and the Research Grants Council of the Hong Kong Special Administrative Region of China (17200119). The computational experiments were performed using research computing facilities offered by Information Technology Services, the University of Hong Kong.

References

Alvarez-Valdes, R., Belenguer, J.M., Benavent, E., Bermudez, J.D., Muñoz, F., Vercher, E., Verdejo, F., 2016. Optimizing the level of service quality of a bike-sharing system. Omega 62, 163–175.

Baker, K.R., 1977. An experimental study of the effectiveness of rolling schedules in production planning. Decis. Sci. 8 (1), 19–27.

Braun, H., 1990. On solving travelling salesman problems by genetic algorithms. In: International Conference on Parallel Problem Solving from Nature, 1990. Springer, Berlin, Heidelberg, pp. 129-133.

Brendel, A.B., Lichtenberg, S., Brauer, B., Nastjuk, I., Kolbe, L.M., 2018. Improving electric vehicle utilization in carsharing: A framework and simulation of an e-carsharing vehicle utilization management system. Transport. Res. D: Transp. Environ. 64, 230–245.

Brennan, J.W., Barder, T.E., 2016. Battery electric vehicles vs. internal combustion engine vehicles. A United States-Based Comprehensive Assessment. [Online] Available: http://www.adlittle.cn/sites/default/files/viewpoints/ADL_BEVs_vs_ICEVs_FINAL_November_292016.pdf. [Accessed June 18, 2020].

Brinkmann, J., Ulmer, M.W., Mattfeld, D.C., 2015. Short-term strategies for stochastic inventory routing in bike sharing systems. Transp. Res. Procedia 10, 364–373.

Brinkmann, J., Ulmer, M.W., Mattfeld, D.C., 2016. Inventory routing for bike sharing systems. Transp. Res. Procedia 19, 316–327.

Brinkmann, J., Ulmer, M.W., Mattfeld, D.C., 2019. Dynamic lookahead policies for stochastic-dynamic inventory routing in bike sharing systems. Comput. Oper. Res. 106, 260–279.

Brinkmann, J., Ulmer, M.W., Mattfeld, D.C., 2020. The multi-vehicle stochastic-dynamic inventory routing problem for bike sharing systems. Business Res. 13 (1), 69–92.

Caggiani, L., Camporeale, R., Ottomanelli, M., Szeto, W.Y., 2018. A modeling framework for the dynamic management of free-floating bike-sharing systems. Transport. Res. C: Emerg. Technol. 87, 159–182.

Caggiani, L., Ottomanelli, M., 2012. A modular soft computing based method for vehicles repositioning in bike-sharing systems. Procedia – Soc. Behav. Sci. 54, 675–684.

Caggiani, L., Ottomanelli, M., 2013. A dynamic simulation based model for optimal fleet repositioning in bike-sharing systems. Procedia – Soc. Behav. Sci. 87, 203–210.

Chiariotti, F., Pielli, C., Zanella, A., Zorzi, M., 2018. A dynamic approach to rebalancing bike-sharing systems. Sensors 18 (2), 512.

Conrad, R.G., Figliozzi, M.A., 2011. The recharging vehicle routing problem. In: Proceedings of the 2011 Industrial Engineering Research Conference, 2011. IISE Norcross, GA, 8.

Contardo, C., Morency, C., Rousseau, L.-M., 2012. Balancing a dynamic public bike-sharing system. Technical Report CIRRELT-2012-09, Montreal, Canada: CIRRELT.

Dantzig, G.B., Ramser, J.H., 1959. The truck dispatching problem. Manage. Sci. 6 (1), 80–91.

Datner, S., Raviv, T., Tzur, M., Chemla, D., 2019. Setting inventory levels in a bike sharing network. Transport. Sci. 53 (1), 62–76.

Desaulniers, G., Errico, F., Irnich, S., Schneider, M., 2016. Exact algorithms for electric vehicle-routing problems with time windows. Oper. Res. 64 (6), 1388–1405.

Erdelić, T., Carić, T., 2019. A survey on the electric vehicle routing problem: variants and solution approaches. J. Adv. Transport. 5075671.

Erdoğan, G., Battarra, M., Wolfler Calvo, R., 2015. An exact algorithm for the static rebalancing problem arising in bicycle sharing systems. Eur. J. Oper. Res. 245 (3), 667–679.

Erdoğan, S., Miller-Hooks, E., 2012. A green vehicle routing problem. Transport. Res. Part E: Logist. Transport. Rev. 48 (1), 100–114.

Felipe, Á., Ortuño, M.T., Righini, G., Tirado, G., 2014. A heuristic approach for the green vehicle routing problem with multiple technologies and partial recharges. Transport. Res. E: Logist. Transport. Rev. 71, 111–128.

Ghosh, S., Varakantham, P., Adulyasak, Y., Jaillet, P., 2017. Dynamic repositioning to reduce lost demand in bike sharing systems. J. Artif. Intell. Res. 58, 387–430.

Hiermann, G., Puchinger, J., Ropke, S., Hartl, R.F., 2016. The electric fleet size and mix vehicle routing problem with time windows and recharging stations. Eur. J. Oper. Res. 252 (3), 995–1018.

Ho, S.C., Szeto, W.Y., 2016. GRASP with path relinking for the selective pickup and delivery problem. Expert Syst. Appl. 51, 14–25.

Hof, J., Schneider, M., Goeke, D., 2017. Solving the battery swap station location-routing problem with capacitated electric vehicles using an AVNS algorithm for vehicle-routing problems with intermediate stops. Transport. Res. B: Methodol. 97, 102–112.

Hosny, M.I., Mumford, C.L., 2009. Investigating genetic algorithms for solving the multiple vehicle pickup and delivery problem with time windows. MIC2009, Metaheuristic International Conference, 2009.

Hu, R., Zhang, Z., Ma, X., Jin, Y., 2021. Dynamic rebalancing optimization for bike-sharing system using priority-based MOEA/D algorithm. IEEE Access 9, 27067–27084.

Jia, Y., Zeng, W., Xing, Y., Yang, D., Li, J., 2021. The bike-sharing rebalancing problem considering multi-energy mixed fleets and traffic restrictions. Sustainability 13 (1), 270.

Karaboga, D., 2005. An idea based on honey bee swarm for numerical optimization. Technical report-tr06, Erciyes University, Engineering Faculty, Computer Engineering Department.

Karaboga, D., Gorkemli, B., 2011. A combinatorial artificial bee colony algorithm for traveling salesman problem. In: 2011 International Symposium on Innovations in Intelligent Systems and Applications, 2011. IEEE, pp. 50-53.

Kaspi, M., Raviv, T., Tzur, M., 2014. Parking reservation policies in one-way vehicle sharing systems. Transport. Res. B: Methodol. 62, 35–50.

Keskin, M., Çatay, B., 2016. Partial recharge strategies for the electric vehicle routing problem with time windows. Transport. Res. C: Emerg. Technol. 65, 111–127.

Keskin, M., Çatay, B., 2018. A matheuristic method for the electric vehicle routing problem with time windows and fast chargers. Comput. Oper. Res. 100, 172–188.

Keskin, M., Çatay, B., Laporte, G., 2021. A simulation-based heuristic for the electric vehicle routing problem with time windows and stochastic waiting times at recharging stations. Comput. Oper. Res. 125, 105060.

Kloimüllner, C., Papazek, P., Hu, B., Raidl, G.R., 2014. Balancing bicycle sharing systems: an approach for the dynamic case. European Conference on Evolutionary Computation in Combinatorial Optimization, 2014. Springer, pp. 73-84.

Lahoorpoor, B., Faroqi, H., Sadeghi-Niaraki, A., Choi, S.-M., 2019. Spatial cluster-based model for static rebalancing bike sharing problem. Sustainability 11 (11), 3205.

Legros, B., 2019. Dynamic repositioning strategy in a bike-sharing system; how to prioritize and how to rebalance a bike station. Eur. J. Oper. Res. 272 (2), 740–753.

Levin, M.W., 2017. Congestion-aware system optimal route choice for shared autonomous vehicles. Transport. Res. C: Emerg. Technol. 82, 229–247.

Li, Y., Zheng, Y., Yang, Q., 2018. Dynamic bike reposition: a spatio-temporal reinforcement learning approach. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2018. pp. 1724-1733.

Lu, J., Chen, Y., Hao, J.K., He, R., 2020. The time-dependent electric vehicle routing problem: model and solution. Expert Syst. Appl. 161, 113593.

Luo, H., Kou, Z., Zhao, F., Cai, H., 2019. Comparative life cycle assessment of station-based and dock-less bike sharing systems. Resour. Conserv. Recycl. 146, 180–189.

Macrina, G., Laporte, G., Guerriero, F., Pugliese, L.D.P., 2019. An energy-efficient green-vehicle routing problem with mixed vehicle fleet, partial battery recharging and time windows. Eur. J. Oper. Res. 276 (3), 971–982.

McKenzie, G., 2018. Docked vs. dockless bike-sharing: contrasting spatiotemporal patterns. In: 10th International Conference on Geographic Information Science (GIScience 2018). Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

McKenzie, G., 2019. Spatiotemporal comparative analysis of scooter-share and bike-share usage patterns in Washington, DC. J. Transp. Geogr. 78, 19–28.

Montoya, A., Guéret, C., Mendoza, J.E., Villegas, J.G., 2017. The electric vehicle routing problem with nonlinear charging function. Transport. Res. B: Methodol. 103, 87–110.

Narayanan, S., Chaniotakis, E., Antoniou, C., 2020. Shared autonomous vehicle services: a comprehensive review. Transport. Res. C: Emerg. Technol. 111, 255–293.

Nourinejad, M., Zhu, S., Bahrami, S., Roorda, M.J., 2015. Vehicle relocation and staff rebalancing in one-way carsharing systems. Transport. Res. E: Logist. Transport. Rev. 81, 98–113.

Peeta, S., Mahmassani, H.S., 1995. Multiple user classes real-time traffic assignment for online operations: a rolling horizon solution framework. Transport. Res. C: Emerg. Technol. 3 (2), 83–98.

Pelletier, S., Jabali, O., Laporte, G., 2018. Charge scheduling for electric freight vehicles. Transport. Res. B: Methodol. 115, 246–269.

Pillac, V., Gendreau, M., Guéret, C., Medaglia, A.L., 2013. A review of dynamic vehicle routing problems. Eur. J. Oper. Res. 225 (1), 1–11.

Raviv, T., Kolka, O., 2013. Optimal inventory management of a bike-sharing station. IIE Trans. 45 (10), 1077–1093.

Ross, S.M., 2014. Introduction to Probability Models. Academic Press.

Savelsbergh, M.W.P., Sol, M., 1995. The general pickup and delivery problem. Transport. Sci. 29 (1), 17–29.

Schiffer, M., Walther, G., 2017. The electric location routing problem with time windows and partial recharging. Eur. J. Oper. Res. 260 (3), 995–1013.

Schneider, M., Stenger, A., Goeke, D., 2014. The electric vehicle-routing problem with time windows and recharging stations. Transport. Sci. 48 (4), 500–520.

Schneider, M., Stenger, A., Hof, J., 2015. An adaptive VNS algorithm for vehicle routing problems with intermediate stops. OR Spectrum 37 (2), 353–387.

Schuijbroek, J., Hampshire, R.C., Van Hoeve, W.-J., 2017. Inventory rebalancing and vehicle routing in bike sharing systems. Eur. J. Oper. Res. 257 (3), 992–1004.

Shu, J., Chou, M.C., Liu, Q., Teo, C.-P., Wang, I.-L., 2013. Models for effective deployment and redistribution of bicycles within public bicycle-sharing systems. Oper. Res. 61 (6), 1346–1359.

Shui, C.S., Szeto, W.Y., 2018. Dynamic green bike repositioning problem – a hybrid rolling horizon artificial bee colony algorithm approach. Transport. Res. Part D: Transp. Environ. 60, 119–136.

Szeto, W.Y., Wu, Y., Ho, S.C., 2011. An artificial bee colony algorithm for the capacitated vehicle routing problem. Eur. J. Oper. Res. 215 (1), 126–135.

Tian, Z., Jung, T., Wang, Y., Zhang, F., Tu, L., Xu, C., Tian, C., Li, X.-Y., 2016. Real-time charging station recommendation system for electric-vehicle taxis. IEEE Trans. Intell. Transp. Syst. 17 (11), 3098–3109.

Usama, M., Shen, Y., Zahoor, O., 2019. Towards an energy efficient solution for bike-sharing rebalancing problems: a battery electric vehicle scenario. Energies 12 (13), 2503.

Vogel, P., Greiser, T., Mattfeld, D.C., 2011. Understanding bike-sharing systems using data mining: exploring activity patterns. Procedia-Soc. Behav. Sci. 20, 514–523.

Vogel, P., Neumann Saavedra, B.A., Mattfeld, D.C., 2014. A hybrid metaheuristic to solve the resource allocation problem in bike sharing systems. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 8457, 16–29.

Wang, Y., Huang, Y., Xu, J., Barclay, N., 2017. Optimal recharging scheduling for urban electric buses: a case study in Davis. Transport. Res. E: Logist. Transport. Rev. 100, 115–132.

Wang, Y., Szeto, W.Y., 2018. Static green repositioning in bike sharing systems with broken bikes. Transport. Res. D: Transp. Environ. 65, 438–457.

Wang, Y., Szeto, W.Y., 2021. An enhanced artificial bee colony algorithm for the green bike repositioning problem with broken bikes. Transport. Res. C: Emerg. Technol. 125, 102895.

Xu, M., Meng, Q., 2019. Fleet sizing for one-way electric carsharing services considering dynamic vehicle relocation and nonlinear charging profile. Transport. Res. B: Methodol. 128, 23–49.

Yi, P., Huang, F., Peng, J., 2019. A rebalancing strategy for the imbalance problem in bike-sharing systems. Energies 12 (13), 2578.

Zhang, D., Yu, C., Desai, J., Lau, H.Y.K., Srivathsan, S., 2017. A time-space network flow approach to dynamic repositioning in bicycle sharing systems. Transport. Res. B: Methodol. 103, 188–207.