| | |
|---|---|
| **Titre:** Title: | Data-driven prioritization strategies for inventory rebalancing in bike-sharing systems |
| **Auteurs:** Authors: | Maria Clara Martins Silva, Daniel Aloise, & Sanjay Dominik Jena |
| **Date:** | 2024 |
| **Type:** | Article de revue / Article |
| **Référence:** Citation: | Silva, M. C. M., Aloise, D., & Jena, S. D. (2024). Data-driven prioritization strategies for inventory rebalancing in bike-sharing systems. Omega, 129, 103141 (15 pages). https://doi.org/10.1016/j.omega.2024.103141 |





| | |
|---|---|
| **Titre de la revue:** Journal Title: | Omega (vol. 129) |
| **Maison d'édition:** Publisher: | Elsevier |
| **URL officiel:** Official URL: | https://doi.org/10.1016/j.omega.2024.103141 |
| **Mention légale:** Legal notice: | © 2024 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/bync-nd/4.0/). |

Omega 129 (2024) 103141

Contents lists available at ScienceDirect

Omega

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

# Data-driven prioritization strategies for inventory rebalancing in bike-sharing systems☆

Maria Clara Martins Silva [a], Daniel Aloise [a,*], Sanjay Dominik Jena [b]

[a] Polytechnique Montréal and GERAD, 2500 Chem. de Polytechnique, H3T 1J4, Montréal, Canada
[b] Université du Québec à Montréal and CIRRELT, 405 Rue Sainte-Catherine Est, Montréal, H2L 2C4, Canada

ARTICLE INFO

*Keywords:*
Bike-sharing
Demand prediction
Rebalancing
Inventory management

ABSTRACT

The popularity of bike-sharing systems has constantly increased throughout the recent years. Most of such success can be attributed to their multiple benefits, such as user convenience, low usage costs, health benefits and their contribution to environmental relief. However, satisfying all user demands remains a challenge, given that the inventories of bike-sharing stations tend to be unbalanced over time. Bike-sharing system operators must therefore intervene to rebalance station inventories to provide both available bikes and empty docks to the commuters. Due to limited rebalancing resources, the number of stations to be rebalanced often exceeds the system's rebalancing capacity, especially close to peak hours. As a consequence, operators are forced to manually select a subset of stations that should be prioritized for rebalancing. While most of the literature has concentrated either on predicting optimal station inventories or on the rebalancing itself, the identification of critical stations that should be prioritized for rebalancing has received little attention. Given the importance of this step in current operating practices, we propose three strategies to select the stations that should be prioritized for rebalancing, using features such as the predicted trip demand and the inventory levels at the stations themselves. Two sets of computational experiments aim at evaluating the performance of the proposed prioritization strategies on real-world data from Montreal's bike-sharing system operator. The first set of experiments focuses on both the 2019 and 2020 seasons, each of which exhibits distinct travel patterns given the restrictive measures implemented in 2020 to prevent the spread of COVID-19. One of these strategies significantly improves by reducing the estimated lost demand by up to 65%, while another strategy reduces the estimated number of required rebalancing operations by up to 33% when compared to the prioritization scheme currently in use at the considered bike-sharing system. The second set of experiments evaluates the performance of the proposed strategies when rebalancing decisions are optimized in a rolling horizon planning. The results highlight various benefits of the proposed strategies, which are efficiently solved as transportation problems and improve lost demand over two intuitive baselines.

## 1. Introduction

Demand for bike-sharing systems (BSSs) has constantly increased in recent years due to their various advantages: they are typically simple to use and do not require prior reservation; they have been shown to be an environmentally friendly transportation mode by reducing the number of cars in circulation [1]; and they contribute to a healthy lifestyle [2]. Particularly throughout the COVID-19 pandemic, BSSs were considered a transportation alternative with a particularly low risk of user contamination [3,4].

In this paper, we focus on dock-based BSSs, in which stations are located in different parts of the city, and from which commuters may rent and return bikes. While dock-based systems have several advantages (e.g. users get used to the location of stations and bikes), a main issue is that the station inventories may quickly become unbalanced, i.e., either rental demand cannot be met, given that not a sufficient number of bikes is available, or return demand cannot be met, when the station has no empty docks. In such cases, the commuter must relocate to the nearest station with available bikes or docks in order to rent or return a bike. The inventory imbalance often occurs during rush hours on weekdays, when commuters relocate from their residential areas to the areas they work in the morning and do the return trip in the afternoon [5]. Unmet user demand likely causes user dissatisfaction, which the system operators seek to avoid as best as possible, given

☆ Area: Data-Driven Analytics. This manuscript was processed by Associate Editor Y. He.

* Corresponding author.
*E-mail addresses:* maria-clara.martins-silva@polymtl.ca (M.C.M. Silva), daniel.aloise@polymtl.ca (D. Aloise), jena.sanjay-dominik@uqam.ca (S.D. Jena).

https://doi.org/10.1016/j.omega.2024.103141
Received 13 September 2023; Accepted 29 June 2024
Available online 4 July 2024

that it ultimately reduces the user base as the system's reputation is damaged.

An effective way to fight station inventory imbalances is to redistribute bikes among stations, a process known as *rebalancing*. The literature distinguishes two main types of rebalancing: user-based rebalancing and operator-based rebalancing. The former consists of incentives given to the users in order to return bikes at stations before they become empty [6]. In contrast, operator-based rebalancing is carried out by the BSS operators themselves, typically by dispatching vehicles that relocate bikes between the stations. In this work, we focus on operator-based rebalancing, which has shown to be effective in increasing demand satisfaction [see, e.g. 7] and is the common practice at major BSSs around the world, such as BIXI Montreal, Citi Bike in New York City and Ecobici in Mexico City. Such rebalancing is also a less expensive solution compared to installing more stations or adding more docks to already existing stations [8].

In most dock-based BSSs with operator-based rebalancing, the decision to actively rebalance a station depends on which stations are considered *unbalanced*. Depending on the BSS, the criteria may be different for a station to be categorized as such. For example, at NiceRide (Minneapolis, U.S), a station is considered to be unbalanced when it is either completely empty or completely full [9]. The operators of Vélo'v (Lyon, France) classify a station as unbalanced if the absolute difference between the number of arrivals and departures is larger than the standard deviation of the distribution of these values over all the stations [10]. BIXI Montreal uses *inventory intervals* that establish an acceptable quantity of bikes at each station. Inventory intervals are manually set by BIXI's dispatching team, based on their experience with the station location, intraday demand fluctuation, and the day of the week.

Nonetheless, the rebalancing process itself remains costly, as it accounts for gas, the maintenance of the vehicle fleet, drivers' salaries, etc. In addition, it reduces the favorable impact that bike sharing claims to have on the environment. All considered, having a fleet of vehicles large enough to rebalance all unbalanced stations at every hour is not financially viable for most BSSs, especially during peak hours. According to JCDecaux, a company that offers self-service bikes to different cities around the world, the estimated cost in 2009 to relocate a single bike within a BSSs was about three dollars [7]. A fleet of vehicles available for rebalancing is therefore limited in size and cannot rebalance all unbalanced stations. It becomes imperative that the selection of stations to rebalance is carried out as effectively as possible.

Consider Fig. 1, showing BIXI's (estimated on historical data) maximum hourly rebalancing capacity and the average number of unbalanced stations per hour for weekdays in July and August of 2019 (left) and 2020 (right). Throughout this period, the number of unbalanced stations consistently exceeded BIXI's maximum rebalancing capacity of approximately 46 stations per hour in 2019 and 22 stations per hour in 2020. As a consequence, the operator must select a subset of these stations to be rebalanced.

Ideally, the subset of unbalanced stations should be selected to maximize the number of served future demand requests, which requires an appropriate demand forecast. However, predicting the demand of a BSS is a complex task depending on several factors, such as the weather, the hour of the day, the day of the week, holidays and public events.

Demand prediction in BSS has become particularly challenging in 2020 due to restrictive measures imposed by the governments in response to the COVID-19 pandemic, leading to a large part of the population working from home. Fig. 2 shows the average number of trips and the average number of rebalancing operations as reported by BIXI during the weekday hours of July and August 2019 (left) and 2020 (right). Not only did the number of trips in 2020 decreased considerably with respect to the same period in 2019, but the trip behavior also changed. In 2019, the peak hours occurred right before and right after the working hours, i.e., at 8 a.m. and 5 p.m., respectively. However, this pattern was no longer observed in 2020, resulting in a flatter demand along the day.

Such drastic demand changes as observed from 2019 to 2020 severely affect the subset of stations that become unbalanced over time, complicating the demand forecast, and, in turn, affecting the choice of stations to be rebalanced. As a result, the pattern of BIXI's rebalancing operations changed drastically, reducing their activities by nearly half during the peak hours in 2020. Given that the manual planning is mostly based on the previous experience of the dispatching team, there is a significant risk that manually adjusted rebalancing strategies are ineffective in practice in a different environment of trip demand. This suggests that data-driven strategies that quickly adapt to changing demand hold certain benefits to assist BSS rebalancing operations.

In this paper, we focus on the data-driven selection of a subset of stations that should be considered for rebalancing. To this end, we first propose a model that automatically generates inventory intervals considering both rentals and returns forecast. Thereafter, we propose three strategies to prioritize the unbalanced stations according to their need for rebalancing. These strategies are based on the current inventory levels at the stations and the predicted demand for the next hours. Each strategy aims to tackle a specific issue in order to improve the BSS performance and allows for integrating look-ahead periods into the computed priority score. The first strategy focuses on prioritizing stations that are the most likely to generate lost demand if not rebalanced. The second strategy aims to prioritize stations according to the amount of lost demand that can be avoided if they are immediately rebalanced. The third strategy prioritizes stations according to how unbalanced they are based on their inventory intervals.

The three prioritization strategies are compared by means of two tailored discrete-time simulations to compare the estimated lost demand (i.e., demand that could not be satisfied), the total number of alerts raised each time a station becomes unbalanced, and the number of performed rebalancing operations. In the first set of experiments, the prioritization strategies are compared to a systematic baseline, which emulates the prioritization strategy currently employed at BIXI. Results reveal that one of the proposed strategies is able to reduce the estimated lost demand by 35% for the 2019 season data, and by 65% for the 2020 season data, as compared to the baseline strategy. Our second simulation extends the first by executing single-period rebalancing planning within a rolling horizon framework. While the series of corresponding single-period planning problems can be modeled as a mixed-integer formulation, we show how to efficiently solve such problems as transportation problems, matching vehicles to stations. The performance of the proposed prioritization strategies is compared to two baselines: the first baseline clusters the regions into independent subregions each served by one vehicle within a multi-period planning model; the second baseline is similar to our approach, but considers all stations as opposed to only the subset identified by the prioritization scores. Results indicate that our prioritization strategies have several benefits over the considered baselines, as they reduce the lost demand and are solved within a matter of milliseconds.

As outlined above, selecting a subset of critical stations that is aligned with the total rebalancing capacity is a primary task and concern of many system operators and, as such, can easily be integrated into the company's decision process. All proposed strategies are easy to implement and computationally cheap. They therefore provide an attractive alternative to more elaborate BSS planning approaches, such as those based on optimization models, which tend to be computationally challenging (or even intractable), given that they require to consider all stations within the same model.

The remainder of this paper is organized as follows. Section 2 reviews the most relevant literature in the area of rebalancing and prioritization strategies for bike-sharing systems. Section 3 describes how the inventory intervals are defined so as to serve as input to the prioritization strategies. Section 4 describes the different strategies proposed to score the rebalancing priorities of unbalanced stations,

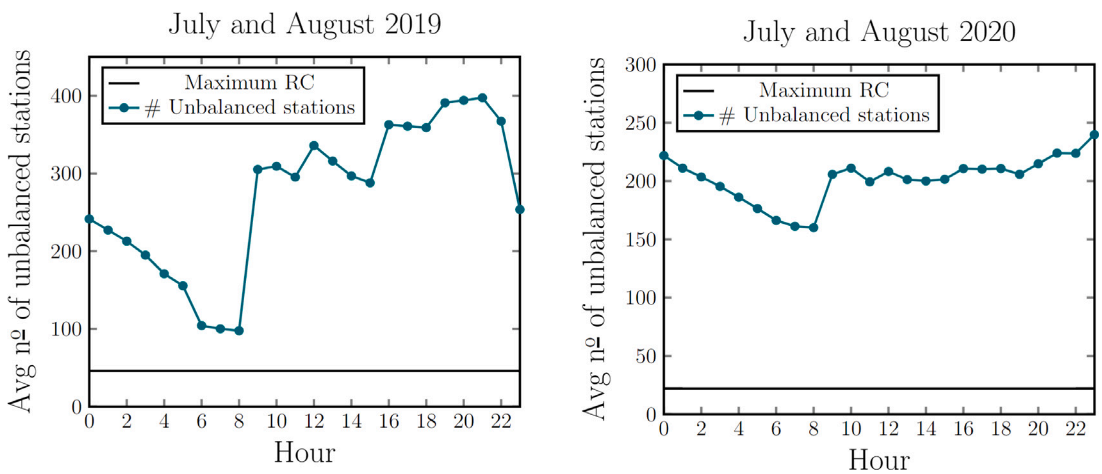


**Fig. 1.** Maximum hourly rebalancing capacity and average number of unbalanced stations at BIXI BSS during weekday hours in July and August 2019 (left) and 2020 (right).

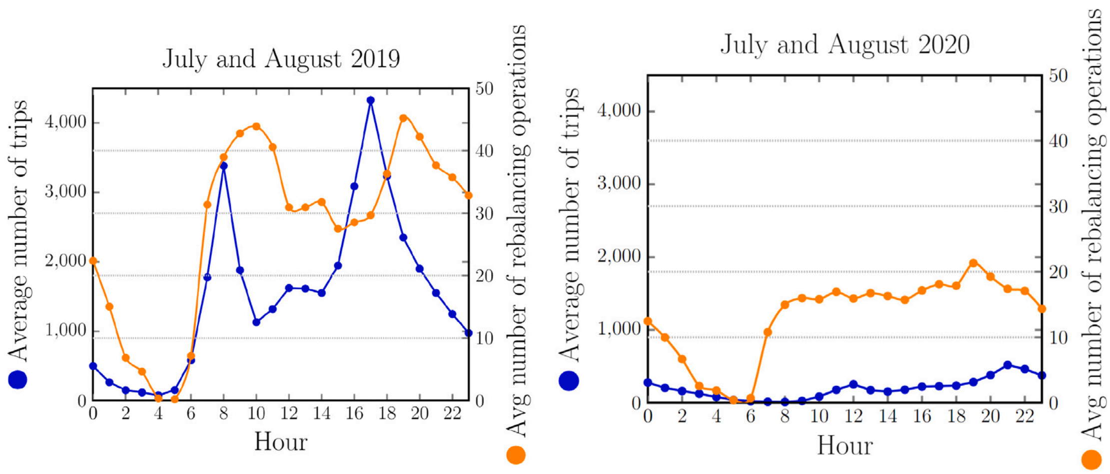


**Fig. 2.** Average number of trips and rebalancing operations for all stations during weekday hours in July and August 2019 (left) and 2020 (right).

provides a theoretical intuition on the benefits of such strategies and shows how to efficiently solve the corresponding planning problem as a transportation problem, i.e., a matching of vehicles to stations. Section 5 presents and analyzes the computational experiments. Finally, Section 6 concludes the paper.

## 2. Literature review

Rebalancing in BSSs can be divided into two main steps: (a) tactical inventory management, and (b) operational bike repositioning. Step (a) aims to establish the number of bikes in each station to meet the predicted demand as best as possible. Step (b) focuses on the actual dispatching operations that are necessary to achieve the desired inventory levels at the stations to rebalance the system.

In order to define, in step (a), the optimal inventories that are likely to provide sufficient bikes and free docks to satisfy future demand, it is necessary to forecast the latter sufficiently well. Trip demand is influenced by numerous external factors, such as the weather, the day of the week, the time of the day, land use, the location of the stations, points of interest, and socio-demographic characteristics [11,12]. Most of the proposed approaches to predict trip demand are either based on machine learning [see, e.g. 13–15] or on statistical models [see, e.g. 10,11,16,17]. These models differ from each other in terms of the predicted time horizon (hourly, daily, or weekly), as well as the geographic granularity of the predictions (station-level, cluster-level, or network-level). For instance, [15,17] predict the total demand in the network for each observed hour. [10,13] predict the total demand for clusters of stations, while [16] estimate the probability that stations in a cluster become either completely full or empty. [11] propose a model that estimates the future demand for each station for five time periods along the day (morning, midday, afternoon, evening, and overnight). [14] predict the rentals and returns for each station per hour, using temporal (day, day of the week, holiday, etc.) and weather (temperature, humidity, rain, etc.) features. The authors also propose a reduction technique for the trip data, improving the computational execution time and erasing outliers from the dataset. For practical purposes, station-level demand predictions for shorter time-periods (such as one hour) seem preferable given that (i) the demand can drastically change from one hour to the next, and that (ii) the rebalancing process is actually planned and carried out at station-level.

Once the demand is properly predicted, optimal target inventory values can be determined. [18] model the station inventory by means of a Poisson queuing system that estimates its optimal number of bikes while ensuring a given service level. Also using a queuing system, [19] compute the target inventory values for the highest-demand stations, denoted central stations, considering their demand as well as those of their nearby stations. In the work of [20], inventories are optimized in order to maximize the amount of time a station is considered balanced. In [21], they estimate the optimum inventory by minimizing a user dissatisfaction function with penalty variables to control the weight given to the rental or return dissatisfaction. Likewise, [14] introduce a hyperparameter to prioritize either the rental or the return service level when computing the target inventory value of a station. [22] obtain target inventory values by minimizing the journey dissatisfaction metric, which is measured by the number of commuters who wait for missing bikes (for rentals) or docks (for returns), or by the number of times they change their desired station. In [23], the authors propose a model that determines the target number of bikes for each zone in the BSS network using a dissatisfaction risk measure. This measure simultaneously quantifies both the probability and the magnitude of user dissatisfaction in a given zone.

Regarding the operational decision-making step (b), the works in the literature can be categorized into two classes: those that assume that rebalancing operations are performed by the users of the system (under some incentive) and those that rebalance by means of a vehicle fleet coordinated by the operator. [24] propose a reward mechanism to encourage users to return bikes to certain stations in the BSS. In [25], the authors conclude that encouraging the users to return the bikes to a non-saturated station does not significantly improve the system's performance. However, they also show that the performance can be improved by constantly stimulating users to return bikes to a nearby station with lower inventory.

In the case of dock-based BSSs, as the ones considered here, operator-based rebalancing via vehicles has typically been modeled via mixed-integer linear programming [see, e.g. 26–37]. These models generally aim to find optimal vehicle routes to rebalance a set of stations, typically seeking to maximize customer satisfaction. The latter may be achieved by minimizing the total lost demand [see, e.g. 26,30,34], minimizing the costs incurred by rebalancing operations [33], keeping station inventories close to their respective target inventory values [see, e.g. 27,36], or even by optimizing several (possibly conflicting) objectives [see, e.g. 38]. Unfortunately, the use of such models in practice is rather challenging, given that the resulting optimization models tend to be hard to solve. This typically limits their use to a small number of stations, given that intraday planning typically requires decisions within a matter of minutes. [18] observe that their formulation becomes difficult to solve even for small instances with 50 stations and 3 vehicles. The authors, therefore, propose a heuristic that clusters stations using a maximum spanning star and then rebalances among clusters. Likewise, [39] cluster nearby stations and then rebalance among the clusters, where the capacity and the inventory of each cluster are given by the sum across its stations. While such an approach improves computational feasibility, it is based on the assumption that nearby stations have similar patterns and that rebalancing within each cluster is time feasible. [40] uses a Markov decision process to prioritize stations segmented in zones, for which a unique vehicle is dedicated for rebalancing their stations. The method also computes the optimal number of bikes that should be added to or moved from the prioritized station. We note that the cited clustering-based approaches work by segmenting the entire network into smaller ones. Consequently, these methods assume that each subproblem is addressed independently by allocating the available vehicles to specific zones.

Several other heuristic methods have been proposed [see, e.g. 41–43]. In particular, [44] propose a large neighborhood search algorithm that optimizes the vehicle routes only for stations that have raised an alert to the system. An interesting characteristic of their optimization model is that such alerts have different priorities which are proportional to their importance in the objective function. The list of stations to rebalance (i.e., those that raised an alert) along with their associated priorities have to be provided as input to the optimization model.

Our proposed prioritization strategies carry out step (a) and a preliminary step to facilitate step (b). While our approach does not directly optimize bike repositioning through the routing of dispatched vehicles, we propose how to reduce the set of stations that should be considered in the network. By focusing on those that are the most urgent, we significantly facilitate efficient rebalancing operations. Indeed, a prior selection of stations can be very useful to scale optimization models to large BSSs by restricting the number of stations to be actually considered. Moreover, such approaches, based on alerts raised for stations that are susceptible to become unbalanced, are also easier to fit into existing practices at several BSSs, which often plan the dispatching operations based on such alerts. Nonetheless, because of limited resources in practice, planners are often required to choose a subset of the stations to rebalance.

The strategies proposed in the next sections seek to recommend the best subset of stations to be rebalanced over a prespecified period of time (e.g. one hour). Given that these prioritization strategies are easy to implement and compute within a matter of seconds, they provide an attractive alternative to computationally expensive multi-period rebalancing optimization models. We reiterate that to the best of our knowledge, there is a lack of existing literature that shares our precise objective of selecting a subset of unbalanced stations with the purpose of making the routing optimization problem computationally tractable while taking into account the entire network.

## 3. Inventory intervals

BSS operators (such as BIXI Montreal) often use intervals of acceptable inventory values, referred to as inventory intervals. They are composed of a lower and an upper bound, as well as a target inventory value, which refers to the ideal inventory value for that station and falls within the lower and upper bounds. Typically, each station has its specific inventory interval defined for a specific time period, and its values may change depending on the hour and day. When the inventory of a station falls outside of its specified inventory interval, the station is classified as unbalanced, which in turn triggers a rebalancing *alert*.

Fig. 3 illustrates the evolution of the inventory of two stations with different inventory intervals and target values. Once the inventory is outside the interval, the station is assumed to be rebalanced such that, at the next hour, the inventory is set back to the target value. The width and the level of the inventory intervals have a major impact on the quantity of lost demand, on the number of raised alerts, and ultimately, on the total number of rebalancing operations. For example, narrow intervals generate more alerts, but tend to keep the inventory closer to its target value, which in turn, decreases the likelihood of lost demand (see Fig. 3, left). In contrast, wide inventory intervals create fewer alerts, but at the expense of keeping the inventory farther from its target values, which may increase the amount of lost demand (see Fig. 3, right). Defining inventory intervals that make this delicate trade-off in order to maximize system performance over the entire day is therefore a challenge.

The model proposed here computes the inventory intervals and the target values based on the *service level* – which is defined as the proportion of satisfied trips at a given station [18]. In the literature, the metric used to estimate user satisfaction can be computed considering either rentals or returns individually [see, e.g. 14,18,19], or, as in our case, in a combined manner that takes both rentals and returns into account together [see, e.g. 21,22].

The inventory of the stations is typically modeled as an M/M/1/K queue, whose parameters are the time between the rentals, the time between the returns, the number of servers (here, a single station), and the maximum capacity of the server (i.e., the total number of docks $C_s$). As often assumed in the literature, we describe the trips using a Poisson distribution [8,14,18,39,45–47]. Hence, the times between rentals and returns follow an exponential distribution. To estimate the rentals and returns rate, we used the gradient-boosted tree model presented by [14]. This model predicts hourly trips at each station of the system and incorporates Singular Value Decomposition (SVD) to reduce dimensionality and exclude outliers.

Given a station $s$ with initial inventory $f$, a time period $[0,T]$, and the parameters mentioned above, we propose to compute the joint service level of $s$ as:

$$SL_s(f,T) = \frac{\int_0^T (\mu_s(t)(1-p_s(f,0,t)) + \lambda_s(t)(1-p_s(f,C_s,t)))dt}{\int_0^T (\mu_s(t)+\lambda_s(t))dt}, \tag{1}$$

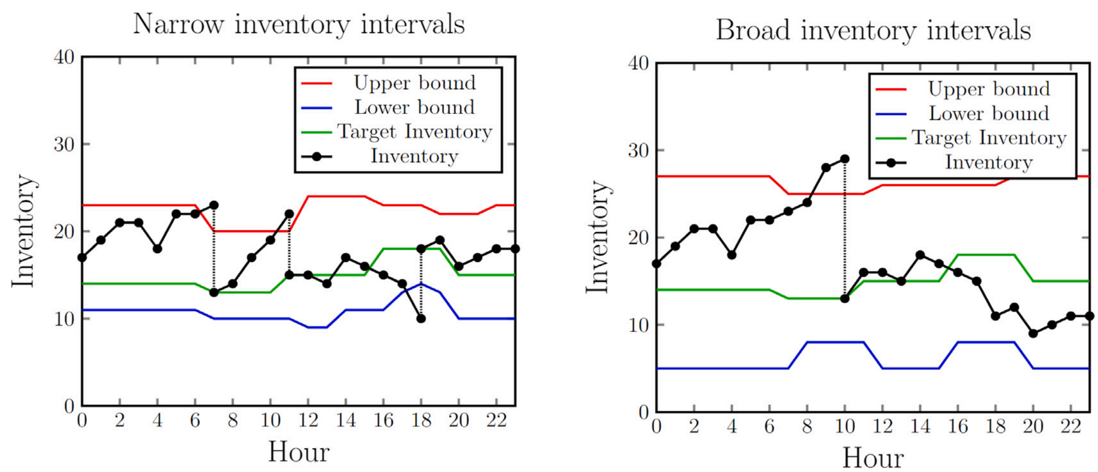


**Fig. 3.** Impact of inventory interval width on inventory fluctuations and triggered rebalancing alerts.

where $p_s(f, N, t)$ is the probability that the station $s$ has $N$ bikes at time $t$, assuming that it had $f$ bikes at time 0, and $\mu_s(t)$ (resp. $\lambda_s(t)$) is the expected rental rate (resp. return rate) during time period $[t, t+1]$ at station $s$. The numerator of Eq. (1) represents the expected number of satisfied trips at station $s$ for the referred period and initial inventory value. After normalization, $SL_s(f, T) \in [0, 1]$.

Thus, we can compute the minimum and maximum service levels for a station $s$ in a time period $[0, T]$ depending on the initial inventory at time 0 as follows:

$$SL_s^{min}(T) = \min_{f \in \{0,\dots,C_s\}} SL_s(f, T), \quad \text{and} \quad SL_s^{max}(T) = \max_{f \in \{0,\dots,C_s\}} SL_s(f, T). \tag{2}$$

Next, we establish a threshold $\Omega_s$ for the acceptable service level for a station $s$ during time period $[0, T]$, defined as:

$$\Omega_s(T) = SL_s^{min}(T) + \beta(SL_s^{max}(T) - SL_s^{min}(T)), \tag{3}$$

where the hyperparameter $\beta \in [0, 1]$ controls how exigent the operator is about the service level. A small value of $\beta$ approximates the threshold to the minimum service level, while a large $\beta$ brings the threshold closer to the maximum service level. Here, $\Omega_s(T) \in [0, 1]$, given that each individual service level $SL_s \in [0, 1]$.

The inventory interval for station $s$ for time period $[0, T]$ compatible with threshold $\Omega_s(T)$ is then defined as

$$\mathcal{I}_s(T) = \{f \in \{0, \dots, C_s\} | \mathcal{L}_s(T) \le f \le \mathcal{U}_s(T)\}, \tag{4}$$

where

$$\mathcal{U}_s(T) = \max\{f \in \{0, \dots, C_s\} | SL_s(f, T) \ge \Omega_s(T)\}, \text{ and} \tag{5}$$

$$\mathcal{L}_s(T) = \min\{f \in \{0, \dots, C_s\} | SL_s(f, T) \ge \Omega_s(T)\}. \tag{6}$$

Finally, the target inventory value $\mathcal{T}_s(T)$ for station $s$ for time period $[0, T]$ is given by the inventory value that provides the highest service level:

$$\mathcal{T}_s(T) = \underset{f \in \{0,\dots,C_s\}}{\arg\max} \{SL_s(f, T)\}. \tag{7}$$

Based on the above defined inventory intervals and the target inventory value, we next propose new strategies to effectively prioritize unbalanced stations that should be considered for rebalancing.

## 4. Prioritization strategies for unbalanced stations

Operators typically follow a systematic approach to prioritize stations for rebalancing among those that have raised an alert. In particular, BIXI Montreal segments the unbalanced stations into three priority groups, each defined by specific criteria.[1] The first group, denoted $K_1$, contains stations classified as critical, i.e., stations that are either completely empty or completely full, and for which all their neighboring stations within a radius of 600 m are also completely empty (or completely full).[2] The second group, denoted $K_2$, includes unbalanced stations that are not part of $K_1$, but are located within a 600-meter radius of a metro station. This criterion recognizes the fact that stations located near metro stations tend to have a higher demand since it is common for both modes of transportation to complement each other in the urban environment. The third group, $K_3$, comprises stations that are not part of the previous groups but are within 600 m of any station in $K_1$ or $K_2$. Any unbalanced station not classified under $K_1, K_2$ or $K_3$ is not considered for rebalancing in the analyzed time period. The priority of stations in $K_1$ is always higher than that of stations in $K_2$, and the priority of stations in $K_2$ is higher than that of stations in $K_3$. Finally, within each group, stations are sorted based on their proximity to the nearest metro station. This means that stations closer to the metro have a greater likelihood of being selected for rebalancing.

In the following, we present our proposed prioritization strategies that provide priority scores to the unbalanced stations according to different criteria. We then provide a theoretical intuition that illustrates under which circumstances these strategies will perform well. Finally, we show how the resulting rebalancing planning for the upcoming planning period can be efficiently solved as a transportation problem.

### 4.1. Prioritization strategy based on inventory forecasting

The first prioritization strategy, denoted $Pa_1$, selects stations for rebalancing taking into consideration the current stations inventories, provided as input, and the expected demand, which is obtained by the gradient-boosted tree regression model of [14].

By considering the current hour $t$, we predict the inventory $\overline{f}_s^1(t')$ of each station $s$ at the beginning of hour $t' \in \{t+1, \dots, t+H\}$, for $H \ge 1$, as:

$$\overline{f}_s^1(t') = \begin{cases} \min\{C_s, \overline{f}_s^1(t'-1) + \lambda_s(t'-1) - \mu_s(t'-1)\}, & \text{if } \lambda_s(t'-1) > \mu_s(t'-1) \\ \max\{0, \overline{f}_s^1(t'-1) + \lambda_s(t'-1) - \mu_s(t'-1)\}, & \text{otherwise.} \end{cases} \tag{8}$$

[1] This procedure was conceived after several exchanges with BIXI's planners. As such, it is not an official representation of BIXI's decision-making process.

[2] The radius of 600 m is defined by BIXI based on the fact that an average person may walk this distance within 10-15 min, which is the time that BIXI considers acceptable for a commuter to walk seeking to be served.

Parameters $\mu_s(t)$ and $\lambda_s(t)$ are, respectively, the expected demand for the number of rentals and returns at station $s$ during time period $[t, t+1]$. In the base case of the recursion (8), where $t' = t+1$, $\overline{f}^1_s(t'-1)$ refers to the actual number of bikes available at station $s$ at the beginning of the current hour $t$.

Then, for each station $s$, the strategy computes for $t' \in \{t+1, \dots, t+H\}$:

$$\mathcal{B}^1_s(t') = \max\{0, -\mathcal{D}^1_s(t'), \mathcal{D}^1_s(t') - C_s\}, \tag{9}$$

where $\mathcal{D}^1_s(t') = \overline{f}^1_s(t'-1) + \lambda_s(t'-1) - \mu_s(t'-1)$. Thus, $\mathcal{B}^1_s(t')$ may indicate either: no lost demand (max 0), a shortfall (max $-\mathcal{D}^1_s(t')$), or a bike surplus (max $\mathcal{D}^1_s(t') - C_s$) during the time period $[t'-1, t]$ at station $s$.

Finally, a prioritization score is computed for station $s$ at hour $t$, proportional to its predicted inventory shortfall or surplus over the period $[t, \dots, t+H]$, as follows:

$$\mathcal{P}^1_s(t) = \sum_{h=1}^{H} \left[1 - \rho \frac{(h-1)}{H}\right] \times \mathcal{B}^1_s(t+h). \tag{10}$$

We note that the terms of (10) are weighted so as to penalize $\mathcal{B}^1_s$ as $h$ increases. The parameter $\rho \in [0,1]$ is thus used to control the rate of discount of the weighting schema over time.

*Theoretical intuition on the impact of the look-ahead period.* Prioritizing stations for rebalancing based solely on the lost demand of the upcoming time-period is likely to lead to higher cumulative lost demand over time, especially when the number of available vehicles for rebalancing is significantly smaller than the number of stations that need rebalancing. Eq. (10) allows for taking into consideration the impact of immediate rebalancing decisions on future time-periods. We illustrate this benefit by means of a toy example:

- Station 1: estimated to have 1 lost rental in the next hour; high rental demand is expected in the upcoming hours during the demand peak.
- Station 2: estimated to have 2 lost rentals in the next hour; rental demand in upcoming hours is expected to remain modest.

When using a myopic look-ahead period of $H = 1$, the prioritization strategy is likely to prioritize station 2, which exhibits an immediately higher number of lost rentals. When using a longer look-ahead period $H > 1$, the anticipated lost rental of the upcoming demand peak can be taken into consideration, therefore prioritizing station 1. Given the limited number of vehicles to rebalance unbalanced stations, a less myopic prioritization strategy would likely outperform a myopic selection of stations. The same reasoning holds for the prioritization strategy introduced next.

### 4.2. Prioritization strategy based on inventory forecasting with immediate rebalancing

The second proposed strategy, denoted $Pa_2$, sorts unbalanced stations according to the amount of lost demand avoided by rebalancing operations. It assumes that a rebalancing operation at the beginning of hour $t$ sets the inventory of a station to its target inventory value.

Thus, the predicted inventory $\overline{f}^2_s(t')$ of each station $s$ at the beginning of hour $t' \in \{t+1, \dots, t+H\}$, for $H \geq 1$, is computed as

$$\overline{f}^2_s(t') = \begin{cases} \min\{C_s, \overline{f}^2_s(t'-1) + \lambda_s(t'-1) - \mu_s(t'-1)\}, & \text{if } \lambda_s(t'-1) > \mu_s(t'-1) \\ \max\{0, \overline{f}^2_s(t'-1) + \lambda_s(t'-1) - \mu_s(t'-1)\}, & \text{otherwise.} \end{cases} \tag{11}$$

Eq. (11) resembles (8), differing only in its base case where $t' = t+1$, for which $\overline{f}^2_s(t'-1)$ represents the target inventory level of station $s$ at hour $t$, denoted by $\mathcal{T}_s(t)$.

The predicted lost demand at station $s$, for $t' \in \{t+1, \dots, t+H\}$, is then computed as:

$$\mathcal{B}^2_s(t') = \max\{0, -\mathcal{D}^2_s(t'), \mathcal{D}^2_s(t') - C_s\}, \tag{12}$$

where $\mathcal{D}^2_s(t') = \overline{f}^2_s(t'-1) + \lambda_s(t'-1) - \mu_s(t'-1)$.

Finally, the prioritization score of station $s$ at hour $t$ is given by strategy $Pa_2$ as:

$$\mathcal{P}^2_s(t) = \mathcal{P}^1_s(t) - \sum_{h=1}^{H} \left[1 - \rho \frac{(h-1)}{H}\right] \times \mathcal{B}^2_s(t+h). \tag{13}$$

The first term of the subtraction in (13), i.e., $\mathcal{P}^1_s(t)$, represents the lost demand if station $s$ is not rebalanced at hour $t$, whereas the second term, refers to the lost demand if that station has its inventory set to its target value $\mathcal{T}_s(t)$ at the beginning of hour $t$. Note that, theoretically, $\mathcal{P}^2_s(t)$ may be negative, i.e., rebalancing a station may increase the lost demand. Throughout our computational experiments with real-world data, however, we have never observed negative values for $\mathcal{P}^2_s(t)$. Finally, the strategy outputs the stations in non-increasing order of their prioritization scores.

*Theoretical intuition on the performance of $Pa_1$ and $Pa_2$.* The two prioritization strategies $Pa_1$ and $Pa_2$ measure the expected lost demand before and after rebalancing, respectively. As these approaches are based on predicted rental and return demand, their performance depends on the quality of the demand predictor. As such, a theoretical guarantee of their capacities to ultimately reduce lost demand is not possible. We may, however, provide an intuition under which conditions one strategy should, theoretically, perform well. If rental and return demand estimates are accurate, those approaches are likely to draw an accurate picture of the station rebalancing priorities. In other words, when rental and return uncertainty is sufficiently small and can be accurately forecasted, strategies $Pa_1$ and $Pa_2$ are expected to perform well since they select stations based on their expected lost demand. $Pa_2$ should be able to accurately indicate the benefits of rebalancing a station since it evaluates the lost demand after a rebalancing operation. In contrast, if rental and return demand is considered rather uncertain, $Pa_1$ and $Pa_2$ may ill estimate the priorities of stations, especially in situations where many stations indicate the same quantity of lost demand. For example, two stations may both have the same estimated quantity of lost rental demand. One of the stations may have an empty inventory, while the other may have a non-empty inventory. In this case, the former should be prioritized. However, exclusively using the lost demand as an evaluation criterion does not allow for distinguishing these stations.

### 4.3. Prioritization strategy using inventory intervals

The third prioritization strategy we propose, denoted $Pa_3$, is more conservative than the strategies above. Instead of giving a high prioritization score to stations whose inventories are predicted below zero or above their dock capacity (i.e., yielding lost demand), $Pa_3$ prioritizes unbalanced stations according to the predicted deviation of the station's inventory from its inventory interval bounds (Eqs. (5)–(6)). Thus, stations might be rebalanced before they actually start to generate lost demand (i.e., before they are completely full or empty).

By considering the current hour $t$, strategy $Pa_3$ computes, for $t' \in \{t+1, \dots, t+H\}$:

$$\mathcal{B}^3_s(t') = \max\{0, \mathcal{L}_s(t') - \mathcal{D}^1_s(t'), \mathcal{D}^1_s(t') - \mathcal{U}_s(t')\}, \tag{14}$$

where $\mathcal{L}_s(t')$ and $\mathcal{U}_s(t')$ correspond, respectively, to the lower and upper bounds of the inventory interval of station $s$ at hour $t'$, and $\mathcal{D}^1_s(t')$ computed as in Section 5.2.1.

The priority score for strategy $Pa_3$ is then computed as:

$$\mathcal{P}^3_s(t) = \sum_{h=1}^{H} \left[1 - \rho \frac{(h-1)}{H}\right] \times \mathcal{B}^3_s(t+h). \tag{15}$$

As a result, $Pa_3$ prioritizes stations that are expected to remain unbalanced in the next hours if the operator does not rebalance them. These stations are sorted and prioritized in non-increasing order of the deviation between their predicted inventories and their inventory interval bounds.

*Theoretical intuition on the performance of $Pa_3$.* While $Pa_1$ and $Pa_2$ prioritize stations based on estimated lost demand, $Pa_3$ provides a safety buffer to the inventory by aiming at an inventory target level that will withstand certain demand uncertainty. The former two strategies are likely to perform well when demand can be accurately predicted. In this situation, strategy $Pa_3$, using an inventory interval, seems unnecessarily conservative. Indeed, the use of an inventory target aims at mitigating demand uncertainty, allowing for prioritizing the station that, currently, is less prepared for the upcoming demand uncertainty. In practice, demand is considered highly uncertain, which suggests that the use of $Pa_3$ may perform better in practice, as long as target intervals are computed appropriately.

### 4.4. Single-period planning used within rolling horizon framework

The proposed prioritization strategies identify a subset of stations that can be considered for immediate rebalancing in the upcoming planning period. While it is difficult to compute such a subset for time periods that lie further ahead, it can nonetheless serve as the basis for executing single-period rebalancing plans within a rolling horizon framework.

The recommendations of station priorities provided by the strategies are made without any knowledge of the available rebalancing resources (typically, vehicles). Their benefits, ignoring the allocation of vehicles to the stations, are empirically evaluated in Section 5.2. To provide efficient rebalancing planning, however, vehicles must be allocated to stations considering current vehicle inventories and their potential to rebalance the inventory of the station they are allocated to. Next, we define such a planning problem and show how to solve it efficiently.

We assume to have available a fleet $V$ of vehicles which are ready for immediate rebalancing, i.e., at the beginning of time $t$. A subset $S' \subseteq S$, denotes the subset of all stations $S$ that should be considered for immediate rebalancing. The expected rental demand for the current time period $[t, t+1]$ is estimated to be $\mu_s(t)$, while the expected return demand is $\lambda_s(t)$. We assume that each vehicle can be sent immediately to any of those stations in $S'$ and carry out rebalancing immediately before such demand occurs, in an effort to better serve such demand. As such, we also assume that the current locations of vehicles do not matter and that the relocation distance is sufficiently small. The capacity of station $s$ is given by $C_s$, while the capacity of vehicle $v$ is given by $\hat{C}_v$. The current inventory of station $s$ is given by $f_s(t)$, whereas the current inventory of vehicle $v$ is given by $\hat{f}_v(t)$. The planning problem considered here requires allocating vehicles $v \in V$ to stations $s \in S'$ such that the demand for the upcoming time period is best served. If $|V| < |S'|$, then not all stations can be served. If $|V| > |S'|$, then not all vehicles will be used. In other words, at most one vehicle is allocated to at most one station.

When using our prioritization strategies to derive set $S'$ for the time period $[t, t+1]$, we aim to rebalance a station inventory as close as possible to the proposed inventory target value $\mathcal{T}_s(t)$. If such values are not available, the problem may aim at directly minimizing the estimated lost demand. In both cases, the problem can be solved as a transportation problem minimizing allocation costs, defined as a bipartite graph with offer nodes $V$ and demand nodes $S'$, where each vehicle has to be assigned to a station. Each offer node has a supply of 1 unit, and each demand node requests 1 unit. If $|V| > |S'|$, artificial station nodes can be added with costs $+\infty$ to ensure that the graph is balanced. In a similar fashion, if $|V| < |S'|$, artificial vehicle nodes can be added with costs $+\infty$ to ensure that the graph is balanced.

Costs on the arcs are computed depending on the objective. If the objective is to minimize the deviation from the inventory target values, costs $c_{v,s}$ for an assignment of vehicle $v$ to station $s$ should generally represent the shortfall of desired rebalancing, i.e., the difference between the target inventory value and the feasible new inventory value after rebalancing. If $\mathcal{T}_s(t) > f_s(t)$, i.e., the vehicle should ideally drop off $\mathcal{T}_s(t) - f_s(t)$ bikes, then the costs should represent the shortfall of the number of bikes that should be dropped off (i.e., $\mathcal{T}_s(t) - f_s(t)$) and the number of bikes that can actually be dropped off, which may be limited by the vehicle inventory $\hat{f}_v(t)$. If $\mathcal{T}_s(t) < f_s(t)$, i.e., the vehicle should ideally pick up $f_s(t) - \mathcal{T}_s(t)$ bikes, the costs should represent the shortfall between the number of bikes that should be picked up (i.e., $f_s(t) - \mathcal{T}_s(t)$) and the number of bikes that can actually be picked up, which may be limited by the space available in the vehicle. As such, the costs of an assignment can be computed as follows:

$$c_{v,s} = \begin{cases} \mathcal{T}_s(t) - f_s(t) - \min\{\mathcal{T}_s(t) - f_s(t), \hat{f}_v(t)\}, & \text{if } \mathcal{T}_s(t) > f_s(t) \\ f_s(t) - \mathcal{T}_s(t) - \min\{f_s(t) - \mathcal{T}_s(t), \hat{C}_v - \hat{f}_v(t)\}, & \text{if } \mathcal{T}_s(t) < f_s(t) \\ +\infty, & \text{if } \mathcal{T}_s(t) = f_s(t). \end{cases} \tag{16}$$

While the costs above represent a shortfall to satisfy, we may simplify the same obtained transportation problem by using costs that represent the reduction of the shortfall:

$$c_{v,s} = \begin{cases} -\min\{\mathcal{T}_s(t) - f_s(t), \hat{f}_v\}, & \text{if } \mathcal{T}_s(t) > f_s(t) \\ -\min\{f_s(t) - \mathcal{T}_s(t), \hat{C}_v - \hat{f}_v\}, & \text{if } \mathcal{T}_s(t) < f_s(t) \\ 0, & \text{if } \mathcal{T}_s(t) = f_s(t). \end{cases} \tag{17}$$

If the objective is to minimize the total lost demand, costs are defined in a similar fashion, but consider the difference between the current station inventory and the net bike outflow $\mathcal{O}_s(t) = \lambda_s(t) - \mu_s(t)$ (which is negative if the rental demand is higher than the return demand in time period $[t, t+1]$). Costs may therefore either represent the shortfall of bikes or docks with respect to the demand, or the reduction of shortfall obtained by rebalancing the station. We here define the latter:

$$c_{v,s} = \begin{cases} -\min\{\mathcal{O}_s(t) - (C_s - f_s(t)), \hat{C}_v - \hat{f}_v\}, & \text{if } \mathcal{O}_s(t) > 0 \ \& \ C_s - f_s(t) < \mathcal{O}_s(t) \\ -\min\{-\mathcal{O}_s(t) - f_s(t), \hat{f}_v(t)\}, & \text{if } \mathcal{O}_s(t) < 0 \ \& \ f_s(t) < -\mathcal{O}_s(t) \\ 0, & \text{otherwise}. \end{cases} \tag{18}$$

The effectiveness of such a single-period rolling horizon planning approach, both in terms of demand satisfaction and computing times, is empirically evaluated in Section 5.3.

## 5. Computational experiments

We now report on computational experiments that have been carried out to evaluate the performance of the proposed prioritization strategies. We first present the details of the dataset used in our study in Section 5.1. The first set of experiments, presented in Section 5.2, focuses on the comparison of several performance measures for the proposed prioritization strategies and the emulated strategy currently in use at BIXI Montreal. We also investigate the impact of hyperparameter $\beta$ used within our strategies, impacting the width of the inventory intervals. Section 5.3 focuses on the second set of experiments, investigating the performance of the strategies when used for single-period planning within a rolling horizon framework. The strategies are compared to two alternative baselines. Also, we investigate the benefits over longer look-ahead periods used within strategies to compute the priority scores.

### 5.1. Data set

We consider real-world data from BIXI Montreal for the 2019 and 2020 seasons. The dataset used in the experiments contains hourly information for time, weather, trips, and stations. Time features hold temporal data such as the hour, day, whether it is a holiday and the day of the week. Weather features store information such as temperature, wind speed and relative humidity. Trip features are composed of the number of bike rentals and returns, observed at each station of the network. Finally, station features contain information regarding the geographical location of each station, as well as their corresponding number of docks. Time and weather features were collected from https://climate.weather.gc.ca (except for the holiday feature that was manually imputed), while trip and station features were both provided by BIXI Montreal. More details about the importance of the different features for the gradient-boosted tree used here can be found in [48].

The BSS network considered here from BIXI Montreal contained a total of 620 stations in 2019 and 641 stations in 2020. Each dataset was split into training, validation, and test data, and they each contain all stations from the corresponding year. The training and validation data are used, respectively, to fit the machine learning model parameters and tune its hyperparameters. Finally, the test dataset was used to provide an unbiased performance evaluation of the different prioritization strategies.

Because of a large observed discrepancy in the frequency and the behavior of trips in 2020, caused by the COVID-19 pandemic, we used different strategies for selecting the dataset for each simulated season. For the 2019 tests, the training dataset contains data from April 2018 to June 2019, minus the months during which BIXI is out of service (i.e., December, January, February, and March). The validation dataset is composed of data from the first 15 days of July and August 2019, and the test dataset uses data from the remaining days of July and August 2019. We opted to divide both July and August into validation and test datasets so that the model is less sensitive to demand changes observed between consecutive months.

Note that the physical network of BSSs typically changes over time, which makes it often difficult to use data linked to a specific station ID over longer periods of time. Therefore, we do not use data prior to the 2018 season. We have also observed that, for the 2020 season, adding training data from previous years deteriorates traffic prediction. Consequently, we used training data from April 2020 to June 2020. In that case, the validation dataset contains the first 15 days of July and August 2020, while the test dataset contains the remaining days of July and August 2020. Note that we focus our experiments on the months of July and August because of their high demand and importance throughout the season.

### 5.2. Comparison with different $\beta$ values for 2019 and 2020

The first set of experiments focuses on comparing our strategies with a systematic approach that emulates the prioritization strategy employed by BIXI Montreal. Here, we use a simple simulator, introduced next, to emulate inventory fluctuations in the BSS according to trip demand and performed rebalancing operations. We then present the computational results under different values of hyperparameter $\beta$ used to define inventory intervals (see Section 3). Specifically, large values of $\beta$ result in narrow inventory intervals, and consequently, in a larger number of alerts, while potentially improving demand satisfaction. In contrast, small $\beta$ values lead to wide intervals, decreasing the number of raised alerts at the stations, but possibly decreasing demand satisfaction.

#### 5.2.1. Simulation

The simulator first initializes the inventories of all stations with their respective target values. Then, at the beginning of each simulated time-period (here, one hour), the subset of unbalanced stations are forwarded for prioritization and, after being sorted, to a post-processing procedure that ensures a balance between the number of bikes picked up and dropped off from the stations (see Appendix A for details). The latter then returns a subset of the stations to be rebalanced. The number of stations in this subset is limited by the total rebalancing capacity (i.e., the vehicle fleet), which is given as input to the simulator and represents the total number of stations the operator is capable of rebalancing at each time-period. In other words, the simple simulator used here prioritizes a number of stations that does not exceed the rebalancing capacity and assumes that each station can be rebalanced to its target value. The inventories of the selected stations are then set to their respective target values. In the sequel, the inventory of all stations is modified according to the historical data of rentals and returns. The simulator then proceeds to the next simulated hour until all hours are iterated.

After execution, the simulator returns three metrics that are important to the BSS operator: (i) the number of raised alerts, indicating how often a station has been classified as unbalanced, thus representing how stressed the system was; (ii) the number of rebalancing operations, allowing for an estimation of the operational rebalancing costs; and (iii) the amount of rental and return requests that could not be satisfied, directly affecting customer satisfaction.

The pseudo-code for this simulation, as well as its execution pipeline and description, can be found at github.com/clara91/Data-driven-prioritization.

#### 5.2.2. Results

We now report on the simulation results with BIXI's data for the 2019 and 2020 seasons. We considered that the maximum rebalancing capacity per hour was 46 for the 2019 season and 22 for the 2020 season. These numbers correspond to the average number of rebalancing operations performed by BIXI during the peak rebalancing hours in the two observed years.

All prioritization strategies have been fed with the same inventory intervals in order to ensure a fair comparison of their performance. Even though the inventory intervals used by BIXI were available to us, incorporating them into the analysis may have compromised the validity of our conclusions. We evaluated the prioritization strategies using three values for $\beta$, specifically $\beta = 0.75$ (narrow intervals), $\beta = 0.50$ (medium intervals), and $\beta = 0.25$ (wide intervals).

*Evaluation of $Pa_1$, $Pa_2$ and $Pa_3$.* Table 1 provides an overview of the total lost demand (expressed as a percentage of the total demand, i.e., rental plus return demand) calculated across the entire network for the simulated hours from the test dataset. The table also includes the hourly average number of raised alerts and the hourly average number of rebalancing operations performed in the system over the same 768 simulated hours. The presented results were collected from the simulation using the proposed prioritization strategies $Pa_1, Pa_2$, and $Pa_3$, as well as the emulation of BIXI's prioritization strategy. Finally, for each of the presented metrics, we report the relative difference (column $\Delta(\%)$) calculated with respect to the values obtained using BIXI's prioritization strategy.

Table 1 suggests the following conclusions:

1. **General impact on lost demand.** The three proposed strategies consistently resulted in lower levels of lost demand compared to BIXI's prioritization strategy across all simulated scenarios. This suggests that the proposed strategies excel at prioritizing stations for rebalancing.
2. **Best strategy to reduce lost demand.** Among the proposed prioritization strategies, $Pa_3$ is the most effective strategy to reduce lost demand and the number of raised alerts, achieving reductions of $\approx$ 35% in lost demand for 2019, and $\approx$ 65% in 2020. However, it is important to consider the tradeoff involved when using $Pa_3$ as it requires a higher number of rebalancing operations compared to BIXI's prioritization strategy. Specifically, even though the performed rebalancing respects the maximum rebalancing capacity, more rebalancing operations also translate into higher operational costs.

**Table 1**
Performance metrics for the 2019 and 2020 seasons.

| Season | Prioritization Strategy | $\beta$ | Lost Demand | | Alerts | | Rebalancing | |
|---|---|---|---|---|---|---|---|---|
| | | | total % | Δ(%) | per hour | Δ(%) | per hour | Δ(%) |
| 2019 | BIXI | 0.25 | 5.22 | | 69.86 | | 23.86 | |
| | | 0.50 | 4.96 | | 95.64 | | 29.48 | |
| | | 0.75 | 5.13 | | 142.41 | | 35.87 | |
| | $Pa_1$ | 0.25 | 4.19 | ▼ −19.72 | 69.34 | ▼ −0.74 | 23.35 | ▼ −2.14 |
| | | 0.50 | 4.05 | ▼ −18.23 | 109.83 | ▲ 14.84 | 23.83 | ▼ −19.17 |
| | | 0.75 | 4.00 | ▼ −22.03 | 170.38 | ▲ 19.64 | 24.02 | ▼ −33.04 |
| | $Pa_2$ | 0.25 | 4.16 | ▼ −20.24 | 69.03 | ▼ −1.19 | 23.20 | ▼ −2.77 |
| | | 0.50 | 4.05 | ▼ −18.35 | 110.01 | ▲ 15.03 | 23.63 | ▼ −19.84 |
| | | 0.75 | 3.95 | ▼ −22.95 | 169.87 | ▲ 19.28 | 23.89 | ▼ −33.40 |
| | $Pa_3$ | 0.25 | 3.99 | ▼ −23.46 | 53.54 | ▼ −23.36 | 27.19 | ▲ 13.96 |
| | | 0.50 | 3.58 | ▼ −27.71 | 74.80 | ▼ −21.79 | 31.51 | ▲ 6.89 |
| | | 0.75 | 3.33 | ▼ −35.13 | 116.63 | ▼ −18.10 | 36.03 | ▲ 0.45 |
| 2020 | BIXI | 0.25 | 2.47 | | 43.60 | | 9.55 | |
| | | 0.50 | 2.13 | | 57.22 | | 11.42 | |
| | | 0.75 | 2.10 | | 88.21 | | 14.59 | |
| | $Pa_1$ | 0.25 | 1.57 | ▼ −36.44 | 31.74 | ▼ −27.20 | 10.93 | ▲ 14.45 |
| | | 0.50 | 1.53 | ▼ −28.17 | 51.98 | ▼ −9.16 | 10.97 | ▼ −3.94 |
| | | 0.75 | 1.53 | ▼ −27.14 | 92.29 | ▲ 4.63 | 10.99 | ▼ −24.67 |
| | $Pa_2$ | 0.25 | 1.55 | ▼ −37.25 | 32.02 | ▼ −26.56 | 10.92 | ▲ 14.35 |
| | | 0.50 | 1.53 | ▼ −28.17 | 51.98 | ▼ −9.16 | 10.97 | ▼ −3.94 |
| | | 0.75 | 1.52 | ▼ −27.62 | 92.65 | ▲ 5.03 | 11.06 | ▼ −24.19 |
| | $Pa_3$ | 0.25 | 1.15 | ▼ −53.44 | 20.64 | ▼ −52.66 | 12.53 | ▲ 31.20 |
| | | 0.50 | 0.88 | ▼ −58.69 | 28.26 | ▼ −50.61 | 14.07 | ▲ 23.20 |
| | | 0.75 | 0.72 | ▼ −65.71 | 48.87 | ▼ −44.60 | 16.47 | ▲ 12.89 |

3. **Best strategies with moderate rebalancing.** $Pa_1$ and $Pa_2$ reduce the lost demand without necessarily increasing the number of rebalancing operations. In fact, they often led to fewer rebalancing operations in the system — for all $\beta$ values in 2019, and 2 out of 3 in 2020. This is explained by the fact that both $Pa_1$ and $Pa_2$ assign a positive score to unbalanced stations only if they are predicted to generate lost demand in the next hour, while $Pa_3$ bases its prioritization score on the violation of their inventory intervals. As a result, $Pa_3$ prioritizes a larger number of stations with scores greater than 0, leading to an increased frequency of rebalancing operations compared to $Pa_1$ or $Pa_2$.
4. **Impact of inventory interval size.** For our prioritization strategies, the relative reduction of the lost demand was more substantial when narrow inventory intervals ($\beta = 0.75$) were employed, generally leading to a higher number of alerted stations and slightly more rebalancing operations. In contrast, this behavior has not been observed during the 2019 season for BIXI's prioritization strategy. Here, narrow inventory intervals ($\beta = 0.75$) resulted in more alerts and rebalancing operations, but this did not translate into a lower lost demand. We analyze this curious behavior below in .
5. **Pattern shift from 2019 to 2020.** As a result of the lockdown measures applied by the Canadian authorities in response to the COVID-19 pandemic, user behavior drastically changed. Most likely attributable to the increased work-from-home, the demand peaks shifted and the total number of trips reduced by 85% (e.g., from about 4000 trips to about 600 trips at the respective peak hours). As a result, the relative lost demand also dropped by at least 50% from 2019 to 2020, as stations were substantially less stressed and users were less often faced with empty or full stations.
6. **Benefits of data-driven methods.** Even though the relative lost demand reduced from 2019 to 2020, this was expected (see previous point) and it is questionable whether the BIXI's rebalancing strategy has adapted to the new demand pattern in an ideal manner. Our prioritization strategies demonstrated a substantially higher reduction of the relative lost demand over BIXI's strategy in 2020. Such an improvement, despite the fact that the total demand was much lower, is remarkable. It is, in fact, much more difficult to further reduce lost demand when the total number of trips is low. For example, 5% of lost demand at a peak hour in 2020 may refer to about 15 out of 600 trips, while it refers to 200 out of 4000 trips in 2019. Reducing an abundant lost demand is easier than a sparse one since the latter requires a very precise identification of the stations at which lost demand can be further reduced. It is therefore even more impressive to see the lost demand in 2019 reducing to about 0.72% for strategy $Pa_3$. This not only highlights that our prioritization strategies are effective, but it also illustrates their ability to adapt to new demand patterns and, as such, highlights the importance of data-driven strategies in general that can adapt to changes in demand patterns much faster than manual adjustments.

In Appendix B, we report the results of Table 1 now segmented according to the distance of the stations to their closest metro station. The results indicate that our prioritization strategies, $Pa_1$, $Pa_2$, and $Pa_3$, exhibit a tendency to distribute rebalancing operations more evenly among *near* and *far* stations compared to BIXI's prioritization strategy (outlined in Section 4). Furthermore, our prioritization strategies generate a greater number of alerts for *near* stations compared to those raised by BIXI's prioritization. In terms of measured lost demand, while $Pa_1$ and $Pa_2$ result in higher lost demand for *near* stations with $\beta$ values of 0.5 and 0.75, $Pa_3$ consistently records lower lost demand for both *near* and *far* stations across all $\beta$ values.

*Trade-off between the number of alerts and the rebalancing capacity.* BIXI's prioritization strategy presenting higher lost demand with tighter inventory intervals ($\beta = 0.75$) is unexpected, given that our prioritization strategies consistently reduced lost demand as the inventory intervals got tighter. It turns out that the trade-off between the number of alerts and the rebalancing capacity is an important one.

Fig. 4 presents the lost demand as a function of the BSS rebalancing capacity for both BIXI's prioritization strategy (left) and $Pa_3$ (right) throughout the 2019 test dataset. For strategy $Pa_3$, more narrow intervals (i.e., higher $\beta$ values) consistently lead to lower lost demand. In contrast, for BIXI's prioritization strategy, narrow intervals only perform well when the rebalancing capacity is sufficiently high (more

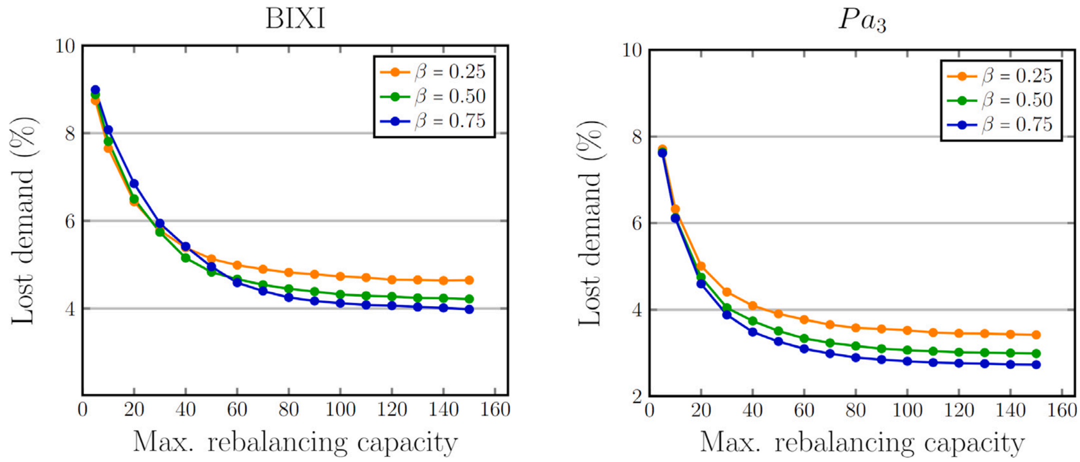


**Fig. 4.** Percentage of lost demand as a function of the rebalancing capacity of the system on the 2019 test dataset.

than 50) to deal with the large amount of raised alerts. When stations are stressed (i.e., there is a high number of raised alerts), BIXI's prioritization strategy has difficulties identifying the most critical stations, which leads to higher lost demand when the rebalancing capacity is low (here, less than 40). Here, wider inventory intervals raise alerts only for the most unbalanced stations, which is more aligned with the limited rebalancing capacity. Indeed, the rebalancing capacity used in our case-study in Table 1 assumes a rebalancing capacity of 46, which lies at the threshold where more narrow intervals are beneficial when using BIXI's prioritization strategy.

This analysis allows for deriving two key insights for BSS operators, particularly those with a limited rebalancing capacity (which is an economical key concern for operators). First, it highlights the importance of an effective prioritization strategy, particularly when the number of alerts is high. Prioritizing the wrong stations is as ineffective as raising the wrong alerts. Second, being capable of adjusting the number of alerts (by smartly selecting the intervals and choosing an appropriate interval size) is crucial, since the operator may want to keep the number of alerts aligned with the rebalancing capacity, particularly when the subsequent prioritization process is not robust to changes in the demand pattern. The prioritization strategies presented here improve upon BIXI's prioritization strategy in both concerns, making them an attractive tool to integrate when working towards more effective and automated rebalancing strategies.

### 5.3. Comparison of different rebalancing strategies

The second set of experiments, presented in this subsection, focuses on the comparison of the various approaches within an optimized rolling horizon planning framework. Even though each of the planning problems corresponds to a single-period problem, our proposed prioritization strategies benefit from a look-ahead period, allowing them to evaluate the decisions' impact on lost demand further in the future. To this end, a more detailed simulator is used, which is introduced next, along with the baseline models. We then present and analyze the computational results.

#### 5.3.1. Baseline models and simulation

We now empirically compare the performance of the prioritization strategies $Pa_1$, $Pa_2$ and $Pa_3$ when used for a single-period rebalancing model within a rolling-horizon framework. We name this model $SP^{TV}_{prior}$, minimizing the deviation from the inventory target values. This model is compared against two baselines:

- $SP^{LD}_{all}$: a single-period rebalancing planning model that minimizes the lost demand on the entire station network.
- $MP_{clusters}$: a geographical clustering approach, where the rebalancing planning for each cluster is computed by solving a multi-period planning model with a single vehicle.

As shown in Section 4.4, both $SP^{TV}_{prior}$ and $SP^{LD}_{all}$ can be solved efficiently as a transportation problem, where the station selection ultimately minimizes the estimated lost demand occurring from a match between vehicles and stations, with costs defined by (17) and (18), respectively. The second baseline, $MP_{clusters}$, relies on a network decomposition approach, given that solving multi-period planning models on the entire network exceeds the capabilities of commercial optimization solvers when dealing with the complete set of stations. $MP_{clusters}$ considers a partition of the stations into $k$ clusters, where $k$ is given by the number of vehicles available for rebalancing. The stations are clustered using the $k$-means algorithm [49], taking into consideration their locations so as to ensure that stations located close to each other belong to the same cluster. Once the clusters are established, a multi-period planning model [50] is solved for each cluster with a single vehicle, utilizing a rolling horizon with 8 time-periods each. At every two time periods, the model is reoptimized, with the solution for the first two time-periods implemented. Note that both baselines consider all stations of the network, whereas $SP^{TV}_{prior}$ only uses a subset of the stations, as prescribed by the corresponding prioritization strategy.

Our simulation starts the inventory of all stations with their respective target values. Then, it iteratively executes the following steps for all simulated time-periods, which are here set to 1 h each:

1. Predict the expected demand and inventory of the unbalanced stations in the network for the upcoming hour(s).
2. Compute inventory intervals and target inventory values.
3. Prioritize unbalanced stations for rebalancing planning (in the case of $SP^{TV}_{prior}$ only).
4. Solve a rebalancing planning model ($SP^{LD}_{all}$, $MP_{clusters}$ or $SP^{TV}_{prior}$) considering the prioritized stations in Step 3, and the current inventory values of stations and vehicles.
5. Perform the resulting planning thereby updating the inventory of the rebalanced stations and used vehicles.
6. Update the inventory of the stations according to rentals and returns data.
7. Compute alerts and update the set of unbalanced stations. Return to Step 1.

The pseudo-code for $SP^{TV}_{prior}$, as well as its description, can also be found at github.com/clara91/Data-driven-prioritization.

**Table 2**
Performance metrics of the rebalancing planning models for the 2019 season.

| Model | $\rho$ | Priorit.Strategy | Lost Demand | | Rebalancing | | Distance | | CPU time |
|---|---|---|---|---|---|---|---|---|---|
| | | | total(%) | Δ(%) | per hour | Δ(%) | per hour | Δ(%) | total |
| $SP^{LD}_{all}$ | – | – | 8.65 | | 13.17 | | 39.17 | | 24.49 |
| $MP_{clusters}$ | – | – | 8.45 | ▼ −2.31 | 10.92 | ▼-17.08 | 20.75 | ▼-47.03 | 46 887.30 |
| $SP^{TV}_{prior}$ | 0 | $Pa_1^{H=1}$ | 7.00 | ▼-19.08 | 10.10 | ▼-23.31 | 31.08 | ▼-20.65 | 1.36 |
| | | $Pa_1^{H=2}$ | 6.89 | ▼-20.35 | 10.77 | ▼-18.22 | 32.61 | ▼-16.75 | 1.41 |
| | | $Pa_1^{H=3}$ | 6.79 | ▼-21.50 | 11.42 | ▼-13.29 | 33.73 | ▼-13.89 | 1.46 |
| | | $Pa_2^{H=1}$ | 6.96 | ▼-19.54 | 9.98 | ▼-24.22 | 30.84 | ▼-21.27 | 1.26 |
| | | $Pa_2^{H=2}$ | 6.85 | ▼-20.81 | 10.54 | ▼-19.97 | 32.06 | ▼-18.15 | 1.42 |
| | | $Pa_2^{H=3}$ | 6.85 | ▼-20.81 | 11.00 | ▼-16.48 | 33.81 | ▼-13.68 | 1.43 |
| | | $Pa_3^{H=1}$ | 6.85 | ▼-20.81 | 12.51 | ▼-5.01 | 34.53 | ▼-11.85 | 1.75 |
| | | $Pa_3^{H=2}$ | 6.90 | ▼-20.23 | 12.68 | ▼-3.72 | 34.87 | ▼-10.98 | 1.61 |
| | | $Pa_3^{H=3}$ | 6.96 | ▼-19.54 | 12.78 | ▼-2.96 | 35.04 | ▼-10.54 | 1.57 |
| | 1 | $Pa_1^{H=1}$ | 7.00 | ▼-19.08 | 10.10 | ▼-23.31 | 31.08 | ▼-20.65 | 1.43 |
| | | $Pa_1^{H=2}$ | 6.84 | ▼-20.92 | 10.77 | ▼-18.22 | 32.32 | ▼-17.49 | 1.27 |
| | | $Pa_1^{H=3}$ | 6.80 | ▼-21.39 | 11.46 | ▼-12.98 | 34.35 | ▼-12.31 | 1.44 |
| | | $Pa_2^{H=1}$ | 6.96 | ▼-19.54 | 9.98 | ▼-24.22 | 30.84 | ▼-21.27 | 1.37 |
| | | $Pa_2^{H=2}$ | 6.85 | ▼-20.81 | 10.68 | ▼-18.91 | 32.92 | ▼-15.96 | 1.27 |
| | | $Pa_2^{H=3}$ | 6.81 | ▼-21.27 | 10.99 | ▼-16.55 | 33.84 | ▼-13.61 | 1.48 |
| | | $Pa_3^{H=1}$ | 6.85 | ▼-20.81 | 12.51 | ▼-5.01 | 34.53 | ▼-11.85 | 1.51 |
| | | $Pa_3^{H=2}$ | 6.87 | ▼-20.58 | 12.45 | ▼-5.47 | 33.92 | ▼-13.40 | 1.56 |
| | | $Pa_3^{H=3}$ | 6.93 | ▼-19.88 | 12.59 | ▼-4.40 | 34.47 | ▼-12.00 | 1.60 |

### 5.3.2. Results

Table 2 presents the results of $SP^{LD}_{all}$, $MP_{clusters}$ and $SP^{TV}_{prior}$ for the 2019 BIXI season data. They consider a rebalancing fleet of 15 vehicles, where $MP_{clusters}$ uses 15 clusters obtained by $k$-means with cardinalities ranging between 7 and 103 stations. Particularly, for the $SP^{TV}_{prior}$, we tested the three proposed prioritization strategies ($Pa_1$, $Pa_2$ and $Pa_3$), with the use of different look-ahead periods ($H \in \{1,2,3\}$), and different discount parameter values ($\rho \in \{0,1\}$). Besides that, $SP^{TV}_{prior}$ is optimized over 20% more prioritized stations than available vehicles. The table indicates, for each approach, the total lost demand (in % of the total demand), the number of rebalancing operations, and the average distance (in km) traveled by the set of vehicles per hour, computed over the entire planning horizon. Note that we test over the last 16 days of July and August 2019, which corresponds to a total of 768 simulated hours. This corresponds to the optimization of 768 planning models (each solved at each hour) for the single-period models $SP^{LD}_{all}$ and $SP^{TV}_{prior}$, and 384 planning models for $MP_{clusters}$. The last column reports the total CPU time (in seconds) spent in the entire simulation. Next to each performance measure, the table shows the percentage improvement $\Delta$ from baseline $SP^{LD}_{all}$ for that specific measure.

By using the estimated rental and return demand from the predictive model for the next time-periods over the entire planning horizon, $SP^{LD}_{all}$ achieves an estimated lost demand of about 8.65%. By casting the problem as a transportation problem, it is solved quite quickly, within less than 30 s. The cluster-wise multi-period model $MP_{clusters}$ presents a slightly lower lost demand of about 8.45%. Even though the model benefits from a multi-period modeling, vehicles are restricted to rebalance locally in each cluster, which limits the overall effectiveness of this approach. In contrast, the average number of rebalancing operations, and especially, the total distance traveled by the vehicles, are much smaller. Nonetheless, the computing time is also much higher, as expected for a multi-period model.

All three proposed prioritization strategies used for $SP^{TV}_{prior}$ allow for a reduction of the lost demand. Here, $\rho$ is evaluated with values 0 and 1, indicating a high and a low discount factor. As expected, this discount factor slightly impacts the performance when higher look-ahead horizons $H$ are used. Indeed, using less myopic information with $H = 2$ and $H = 3$ seems to be beneficial for strategies $Pa_1$ and $Pa_2$. In contrast, this does not seem to be beneficial for $Pa_3$. For all performance measures, our strategies improve between 3% and 25% over baseline $SP^{LD}_{all}$. In comparison with the clustering approach $MP_{clusters}$, model $SP^{TV}_{prior}$ results in greater distances traveled by vehicles regardless of the prioritization strategy employed, while being more effective at decreasing the lost demand. In particular, the computing time of model $SP^{TV}_{prior}$ is low, and much smaller than the time required for solving $MP_{clusters}$. All prioritization strategies are solved within milliseconds, with less than 2 s in total to solve all rebalancing planning models for the entire planning horizon. This makes them an interesting option to be used within a rolling horizon framework where decisions have to be made in real-time.

We next also analyze the hypothetical performance of the various approaches when perfect information is available, as opposed to using a (sufficiently) accurate predictive model (as it has been the case in Table 2). To this end, Table 3 summarizes the performance measures when perfect rental and return information is used within the optimization planning models. While this is, of course, an unrealistic assumption in practice, it allows us to empirically bound the minimal lost demand one may expect and analyze the behavior of the different approaches in such a situation.

Both $SP^{LD}_{all}$ and $MP_{clusters}$ use objective functions that minimize the lost demand. Given that the demand is assumed to be perfectly known, these models are assumed to perform well. Indeed, $SP^{LD}_{all}$ exhibits a quite low lost demand of about 5.72%. The multi-period model $MP_{clusters}$ does not perform as well, which is attributable to its local rebalancing restrictions, allowing vehicles only to rebalance within their respective cluster. We can also observe that, under perfect information, the prioritization strategy $Pa_3$ does not succeed in outperforming the baseline $SP^{LD}_{all}$ regarding lost demand. This is reasonable, given that $Pa_3$ considers inventory intervals, which aim at withstanding uncertain demand variations. Given that in this study, there is no uncertainty, $SP^{TV}_{prior}$ with $Pa_3$ provides an unnecessarily high inventory buffer, as suggested in Section 4.3. Besides, similar to the previous study, $Pa_3$ does not seem to benefit from a look-ahead horizon $H > 1$. $Pa_1$ and $Pa_2$, however, very often benefit from such a look-ahead horizon.

Finally, we observe that $SP^{TV}_{prior}$ with $Pa_2$ consistently outperforms $SP^{LD}_{all}$ for all performance metrics. The use of look-ahead hours with $Pa_2$ also suggests that such less myopic behavior is indeed beneficial, no matter whether perfect information (which, of course, is unrealistic in practice) or a (sufficiently accurate) predictive model is available, supporting the theoretical intuition provided in Section 4.1.

**Table 3**
Performance metrics of the rebalancing planning models for the 2019 season using perfect demand information.

| Model | $\rho$ | Priorit.Strategy | Lost Demand | | Rebalancing | | Distance | | CPU time |
|---|---|---|---|---|---|---|---|---|---|
| | | | total(%) | Δ(%) | per hour | Δ(%) | per hour | Δ(%) | total |
| $SP^{LD}_{all}$ | – | – | 5.72 | | 14.98 | | 45.63 | | 20.73 |
| $MP_{clusters}$ | – | – | 7.72 | ▲34.96 | 13.51 | ▼-9.81 | 22.61 | ▼-50.47 | 21489.21 |
| $SP^{TV}_{prior}$ | 0 | $Pa_1^{H=1}$ | 5.76 | ▲0.70 | 12.65 | ▼-15.55 | 42.81 | ▼-25.81 | 1.50 |
| | | $Pa_1^{H=2}$ | 5.71 | ▼-0.17 | 13.20 | ▼-11.88 | 43.41 | ▼-24.77 | 1.51 |
| | | $Pa_1^{H=3}$ | 5.83 | ▲1.92 | 13.60 | ▼-9.21 | 43.77 | ▼-24.14 | 1.53 |
| | | $Pa_2^{H=1}$ | 5.69 | ▼-0.52 | 12.52 | ▼-16.42 | 42.42 | ▼-26.48 | 1.46 |
| | | $Pa_2^{H=2}$ | 5.58 | ▼-2.45 | 13.02 | ▼-13.08 | 43.92 | ▼-23.88 | 1.49 |
| | | $Pa_2^{H=3}$ | 5.53 | ▼-3.32 | 13.52 | ▼-9.75 | 45.07 | ▼-21.89 | 1.51 |
| | | $Pa_3^{H=1}$ | 6.10 | ▲6.64 | 13.60 | ▼-9.21 | 38.95 | ▼-32.50 | 1.60 |
| | | $Pa_3^{H=2}$ | 6.37 | ▲11.36 | 13.36 | ▼-10.81 | 36.95 | ▼-35.96 | 1.60 |
| | | $Pa_3^{H=3}$ | 6.51 | ▲13.81 | 13.49 | ▼-9.95 | 37.00 | ▼-35.88 | 1.60 |
| | 1 | $Pa_1^{H=1}$ | 5.76 | ▲0.70 | 12.65 | ▼-15.55 | 42.81 | ▼-25.81 | 1.52 |
| | | $Pa_1^{H=2}$ | 5.65 | ▼-1.22 | 13.29 | ▼-11.28 | 44.01 | ▼-23.73 | 1.49 |
| | | $Pa_1^{H=3}$ | 5.61 | ▼-1.92 | 13.81 | ▼-7.81 | 44.47 | ▼-22.93 | 1.52 |
| | | $Pa_2^{H=1}$ | 5.69 | ▼-0.52 | 12.52 | ▼-16.42 | 42.42 | ▼-26.48 | 1.56 |
| | | $Pa_2^{H=2}$ | 5.55 | ▼-2.97 | 13.11 | ▼-12.48 | 43.90 | ▼-23.92 | 1.62 |
| | | $Pa_2^{H=3}$ | 5.45 | ▼-4.72 | 13.61 | ▼-9.15 | 46.07 | ▼-20.16 | 1.50 |
| | | $Pa_3^{H=1}$ | 6.10 | ▲6.64 | 13.60 | ▼-9.21 | 38.95 | ▼-32.50 | 1.65 |
| | | $Pa_3^{H=2}$ | 6.31 | ▲10.31 | 13.46 | ▼-10.15 | 37.63 | ▼-34.78 | 1.58 |
| | | $Pa_3^{H=3}$ | 6.39 | ▲11.71 | 13.52 | ▼-9.75 | 37.26 | ▼-35.42 | 1.67 |

## 6. Concluding remarks

In this paper, we proposed a series of strategies to select the unbalanced stations that should be prioritized for rebalancing in dock-based bike-sharing systems (BSSs). This is an important issue for BSS operators, given that it is neither economically feasible nor ecologically desirable to provide a sufficiently large vehicle fleet in order to rebalance all unbalanced stations at the same time. As such, we aim at improving a step that is present in many BSS operating processes, yet less addressed by the existing literature.

Our prioritization strategies utilize various criteria for selecting stations for rebalancing. These include: forecasting future lost demand ($Pa_1$), assessing the impact of potential rebalancing on station inventory and predicted lost demand ($Pa_2$), and predicting the deviation of station inventories from their inventory intervals ($Pa_3$) – the latter being automatically computed using demand prediction based on historical trips, weather, and temporal data. The proposed strategies also incorporate a look-ahead period, which considers the impact of current decisions on future time periods. As inventory intervals are already being used in practice by BSS operators, the implementation of our prioritization strategies into the practical information and decision process is straightforward. Once the predictive model is trained, inventory intervals and prioritization strategies are computed in a matter of seconds, which is a practical requirement in the ongoing effort to rebalance stations throughout the day.

By comparing the proposed prioritization strategies, we observe that all of them are capable of reducing the occurrences of lost demand. This shows that a judicious choice of stations to be rebalanced has a considerable impact on the service provided and, hence, on customer satisfaction. Among the proposed strategies, $Pa_1$ and $Pa_2$ considerably reduce the number of performed rebalancing operations in most scenarios, whereas $Pa_3$ is the most effective strategy to improve the service level at the stations. Hence, the choice of the prioritization strategy to implement should depend on the objective pursued by the operator.

Our strategies also seem particularly useful during periods in which the demand changes its patterns. The proposed prioritization strategies seem to adapt better to the demand changes in 2020 in comparison to BIXI's prioritization strategy. This outcome is due to the fact that the former is based on demand prediction while the latter is mostly influenced by the location of the stations.

Further numerical experiments embed the prioritization strategies to pre-select subsets of stations for use in a single-period rebalancing model, executed within a rolling horizon framework. The resulting models are compared to two baselines: a multi-period model, in which one vehicle rebalances a specific clustered region, and an approach similar to ours where the entire set of stations is taken into account. The results indicate the benefits of our strategies, outperforming the baselines in terms of lost demand and rebalancing operations. Solved as a transportation problem, such models are solved within a matter of milliseconds, which is particularly attractive for the practical use within real-time planning.

It is noteworthy to mention that, as all our experiments rely on actual trip data, data regarding trips that were not taken due to bike or dock unavailability were not considered. This limitation can be overcome in future research by conducting experiments using synthetic data. This approach will broaden the scope of scenarios one can examine, allowing for a precise quantification of demand losses and thus facilitating a more thorough assessment of the rebalancing recommendations.

Finally, the effectiveness of our prioritization strategies are an interesting example of the beneficial synergies between machine learning and optimization techniques. The general ideas of the prioritization strategies proposed here may be carried over to different application contexts, where planning problems are too complex and may benefit from a reduced search space of demand points that require immediate attention. For example, in post-disaster humanitarian operations, first-response resources are routed by vehicles to beneficiary groups, which have uncertain and varying degrees of urgency. Prioritization strategies similar to those proposed here may be used to prioritize beneficiary groups that require aid the most, taking into consideration the estimated degradation of health condition over a look-ahead time horizon.

## CRediT authorship contribution statement

**Maria Clara Martins Silva:** Writing – review & editing, Writing – original draft, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Daniel Aloise:** Writing – review & editing, Writing – original draft, Supervision, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization. **Sanjay Dominik Jena:** Writing – review & editing, Writing – original draft, Supervision, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization.

## 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.

## Data availability

The authors do not have permission to disclose part of the data.

## Acknowledgments

The authors are grateful to BIXI Montreal who provided resources throughout the project. The authors also thank the Natural Sciences and Engineering Research Council of Canada (NSERC) for its financial support.

## Appendix A. Post-processing to ensure a balanced selection of pick-ups and drop-offs

Our prioritization strategies are responsible for sorting the stations according to their rebalancing priority, but they do not take into consideration practical constraints such as the hourly rebalancing capacity of the operator or the parity between the amount of bikes added and removed from the BSS. Indeed, the latter is an important criterion for the rebalancing team in order to ensure that the number of bikes picked up is approximately the same as the number of bikes dropped off.

In view of that, we developed a post-processing procedure that follows the application of any of the previous prioritization strategies. This procedure has two steps and is illustrated in Fig. A.5. In the first step, the procedure receives the list of sorted unbalanced stations and splits it into two sorted sub-lists of stations, namely: (i) a list $\mathcal{S}^{add}$ of stations where bikes should be added (i.e., dropped off), and (ii) a list $\mathcal{S}^{rem}$ of stations from which bikes should be removed (i.e., picked up). The amount of bikes added or removed from the stations is exemplified in the figure and corresponds to the number of required bikes to restore the inventory of the stations to their respective target values. In the second step, the stations are selected for rebalancing according to the value of the *accumulator* variable, which is initialized with 0. If $accumulator \geq 0$, the procedure selects the highest priority station from $\mathcal{S}^{rem}$ to be rebalanced, and updates the accumulator accordingly. Otherwise, the procedure selects for rebalancing the highest priority station from $\mathcal{S}^{add}$, and the accumulator variable is increased. Each time a station is selected in step 2, it is removed from its corresponding sub-list. The post-processing procedure is halted when either $\mathcal{S}^{add}$ or $\mathcal{S}^{rem}$ are empty, or when the number of rebalancing operations equals the BSS rebalancing capacity (i.e., the maximum number of rebalancing operations the operator is able to carry out). At the end, the procedure outputs the set of stations to be rebalanced in the simulation.

## Appendix B. Prioritization strategies results relative to metro proximity

Table B.4 reports the same results presented in Table 1, but now categorized by station proximity to metro stations. A bike station is classified as *near* for the table if it is at most 600 m away from one metro station. Otherwise, it is classified as *far*.

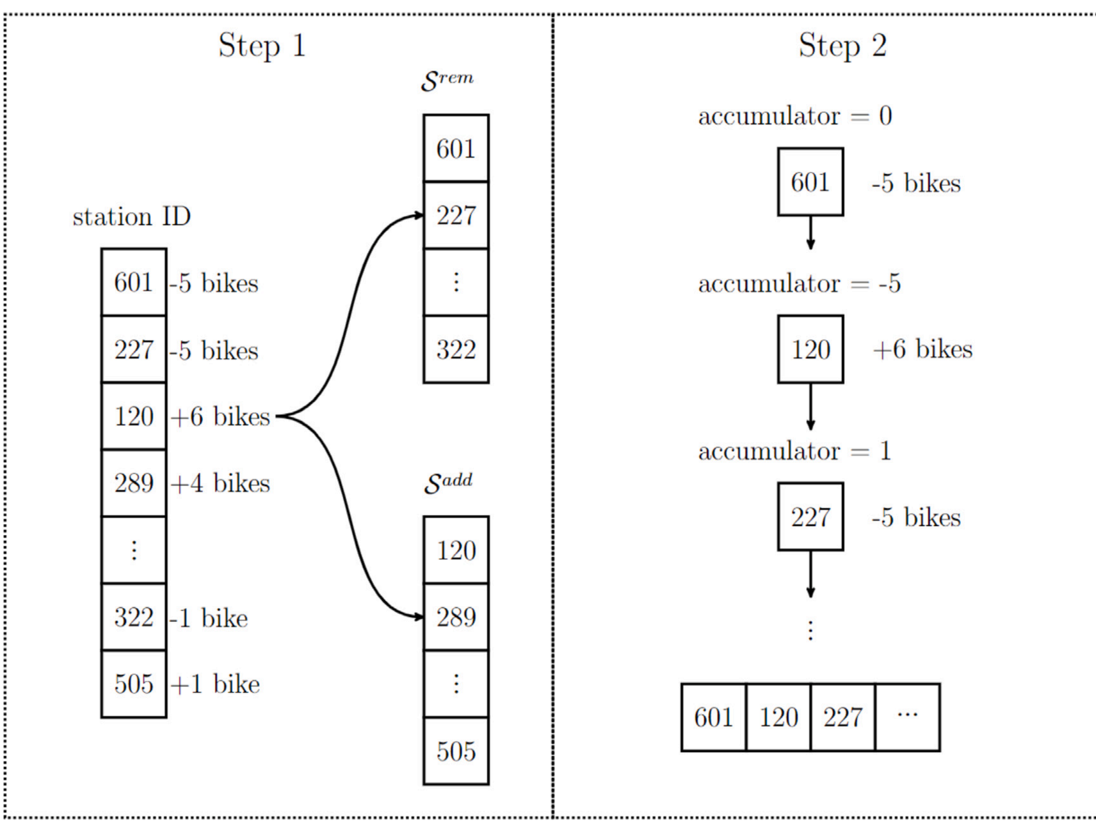


**Fig. A.5.** Illustration of the post-processing procedure divided into two steps. Note that the original sorting of the stations provided by the prioritization strategy is preserved in the sub-lists.

**Table B.4**
Performance metrics for the 2019 season categorized by station proximity to metro stations.

| Season | Strategy | $\beta$ | Lost Demand (%) | | | Alerts per hour | | | Rebalancings per hour | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Near | Far | Total | Near | Far | Total | Near | Far | Total |
| 2019 | BIXI | 0.25 | 4.70 | 6.10 | 5.22 | 29.69 | 40.17 | 69.86 | 20.82 | 3.04 | 23.86 |
| | | 0.50 | 4.36 | 5.96 | 4.96 | 39.27 | 56.37 | 95.64 | 26.19 | 3.29 | 29.48 |
| | | 0.75 | 4.38 | 6.37 | 5.13 | 59.22 | 83.19 | 142.41 | 32.88 | 2.99 | 35.87 |
| | $Pa_1$ | 0.25 | 4.68 | 3.36 | 4.19 | 42.82 | 26.51 | 69.34 | 15.40 | 7.95 | 23.35 |
| | | 0.50 | 4.53 | 3.27 | 4.05 | 67.20 | 42.63 | 109.83 | 15.74 | 8.09 | 23.83 |
| | | 0.75 | 4.49 | 3.19 | 4.00 | 103.30 | 67.08 | 170.38 | 15.89 | 8.13 | 24.02 |
| | $Pa_2$ | 0.25 | 4.65 | 3.33 | 4.16 | 42.66 | 26.37 | 69.03 | 15.33 | 7.87 | 23.20 |
| | | 0.50 | 4.53 | 3.26 | 4.05 | 67.09 | 42.92 | 110.01 | 15.59 | 8.04 | 23.63 |
| | | 0.75 | 4.44 | 3.15 | 3.95 | 103.20 | 66.67 | 169.87 | 15.79 | 8.10 | 23.89 |
| | $Pa_3$ | 0.25 | 4.48 | 3.16 | 3.99 | 32.87 | 20.67 | 53.54 | 17.85 | 9.34 | 27.19 |
| | | 0.50 | 4.05 | 2.82 | 3.58 | 45.34 | 29.46 | 74.80 | 20.67 | 10.84 | 31.51 |
| | | 0.75 | 3.78 | 2.58 | 3.33 | 69.59 | 47.07 | 116.63 | 23.63 | 12.40 | 36.03 |

## References


[1] Wang M, Zhou X. Bike-sharing systems and congestion: Evidence from US cities. J Transp Geogr 2017;65:147–54.
[2] Pucher J, Buehler R, Bassett DR, Dannenberg AL. Walking and cycling to health: a comparative analysis of city, state, and international data. Am J Public Health 2010;100(10):1986–92.
[3] Pase F, Chiariotti F, Zanella A, Zorzi M. Bike sharing and urban mobility in a post-pandemic world. IEEE Access 2020;8:187291–306.
[4] Basak E, Al Balawi R, Fatemi S, Tafti A. When crisis hits: Bike-sharing platforms amid the Covid-19 pandemic. Plos One 2023;18(4):e0283603.
[5] Mellou K, Jaillet P. Dynamic resource redistribution and demand estimation: An application to bike sharing systems. SSRN Electron 2019;1–58.
[6] Vallez CM, Castro M, Contreras D. Challenges and opportunities in dock-based bike-sharing rebalancing: a systematic review. Sustainability 2021;13(4):1829.
[7] DeMaio P. Bike-sharing: History, impacts, models of provision, and future. J Public Transp 2009;12(4):3.
[8] Shu J, Chou MC, Liu Q, Teo C-P, Wang I-L. Models for effective deployment and redistribution of bicycles within public bicycle-sharing systems. Oper Res 2013;61(6):1346–59.
[9] Wang S, He T, Zhang D, Shu Y, Liu Y, Gu Y, et al. BRAVO: Improving the rebalancing operation in bike sharing with rebalancing range prediction. Proc ACM Interact Mob Wearable Ubiquitous Technol 2018;2(1):1–22.
[10] Borgnat P, Abry P, Flandrin P, Robardet C, Rouquier J-B, Fleury E. Shared bicycles in a city: A signal processing and data analysis perspective. Adv Complex Syst 2011;14(03):415–38.
[11] El-Assi W, Salah Mahmoud M, Nurul Habib K. Effects of built environment and weather on bike sharing demand: a station level analysis of commercial bike sharing in Toronto. Transportation 2017;44(3):589–613.
[12] Hampshire RC, Marla L. An analysis of bike sharing usage: Explaining trip generation and attraction from observed demand. In: 91st annual meeting of the transportation research board, washington, DC. 2012, p. 12–2099.
[13] Feng S, Chen H, Du C, Li J, Jing N. A hierarchical demand prediction method with station clustering for bike sharing system. In: 2018 IEEE third international conference on data science in cyberspace. DSC, IEEE; 2018, p. 829–36.
[14] Hulot P, Aloise D, Jena SD. Towards station-level demand prediction for effective rebalancing in bike-sharing systems. In: Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining. 2018, p. 378–86.
[15] Yin Y-C, Lee C-S, Wong Y-P. Demand prediction of bicycle sharing systems. 2012, [Online].
[16] Chen L, Zhang D, Wang L, Yang D, Ma X, Li S, et al. Dynamic cluster-based over-demand prediction in bike sharing systems. In: Proceedings of the 2016 ACM international joint conference on pervasive and ubiquitous computing. 2016, p. 841–52.
[17] Gebhart K, Noland RB. The impact of weather conditions on bikeshare trips in Washington, DC. Transportation 2014;41(6):1205–25.
[18] Schuijbroek J, Hampshire RC, Van Hoeve W-J. Inventory rebalancing and vehicle routing in bike sharing systems. European J Oper Res 2017;257(3):992–1004.
[19] Huang J, Sun H, Li H, Huang L, Li A, Wang X. Central station-based demand prediction for determining target inventory in a bike-sharing system. Comput J 2022;65(3):573–88.
[20] Liu J, Sun L, Chen W, Xiong H. Rebalancing bike sharing systems: A multi-source data smart optimization. In: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. 2016, p. 1005–14.
[21] Raviv T, Kolka O. Optimal inventory management of a bike-sharing station. IIE Trans 2013;45(10):1077–93.
[22] Datner S, Raviv T, Tzur M, Chemla D. Setting inventory levels in a bike sharing network. Transp Sci 2019;53(1):62–76.
[23] Chen Q, Fu C, Zhu N, Ma S, He Q-C. A target-based optimization model for bike-sharing systems: From the perspective of service efficiency and equity. Transp Res B: Methodol 2023;167:235–60.
[24] Chemla D, Meunier F, Pradeau T, Calvo RW, Yahiaoui H. Self-service bike sharing systems: simulation, repositioning, pricing. Technical Report hal-00824078f, Centre d'Enseignement et de Recherche en Mathématiques et Calcul Scientifique - CERMICS; 2013.
[25] Fricker C, Gast N. Incentives and redistribution in homogeneous bike-sharing systems with stations of finite capacity. Euro J Transp Logist 2016;5(3):261–91.
[26] Alvarez-Valdes R, Belenguer JM, Benavent E, Bermudez JD, Muñoz F, Vercher E, et al. Optimizing the level of service quality of a bike-sharing system. Omega 2016;62:163–75.
[27] Brinkmann J, Ulmer MW, Mattfeld DC. Inventory routing for bike sharing systems. Transp Res Proc 2016;19:316–27.
[28] Bulhões T, Subramanian A, Erdoğan G, Laporte G. The static bike relocation problem with multiple vehicles and visits. European J Oper Res 2018;264(2):508–23.
[29] Chemla D, Meunier F, Calvo RW. Bike sharing systems: Solving the static rebalancing problem. Discrete Optim 2013;10(2):120–46.
[30] Contardo C, Morency C, Rousseau L-M. Balancing a dynamic public bike-sharing system. vol. 4, CIRRELT Montreal; 2012.
[31] Dell'Amico M, Hadjicostantinou E, Iori M, Novellani S. The bike sharing rebalancing problem: Mathematical formulations and benchmark instances. Omega 2014;45:7–19.
[32] Erdoğan G, Battarra M, Calvo RW. An exact algorithm for the static rebalancing problem arising in bicycle sharing systems. European J Oper Res 2015;245(3):667–79.
[33] Li X, Wang X, Feng Z. Dynamic repositioning in bike-sharing systems with uncertain demand: An improved rolling horizon framework. Omega 2024;103047.
[34] Lowalekar M, Varakantham P, Ghosh S, Jena S, Jaillet P. Online repositioning in bike sharing systems. In: Proceedings of the international conference on automated planning and scheduling. Vol. 27, 2017, p. 200–8.
[35] Pal A, Zhang Y. Free-floating bike sharing: Solving real-life large-scale static rebalancing problems. Transp Res C 2017;80:92–116.
[36] Papazek P, Raidl GR, Rainer-Harbach M, Hu B. A PILOT/VND/GRASP hybrid for the static balancing of public bicycle sharing systems. In: International conference on computer aided systems theory. Springer; 2013, p. 372–9.
[37] Yu L, Xu Y, Shi H, Zhang Z. Robust optimization model based on a static rebalancing design for bike-sharing systems affected by demand uncertainty. IEEE Access 2024.
[38] Nunes P, Moura A, Santos J. Solving the multi-objective bike routing problem by meta-heuristic algorithms. Int Trans Oper Res 2023;30(2):717–41.
[39] Ghosh S, Varakantham P, Adulyasak Y, Jaillet P. Dynamic repositioning to reduce lost demand in bike sharing systems. J Artificial Intelligence Res 2017;58:387–430.
[40] Legros B. Dynamic repositioning strategy in a bike-sharing system; how to prioritize and how to rebalance a bike station. European J Oper Res 2019;272(2):740–53.
[41] Lu Y, Benlic U, Wu Q. An effective memetic algorithm for the generalized bike-sharing rebalancing problem. Eng Appl Artif Intell 2020;95:103890.
[42] Papazek P, Kloimüllner C, Hu B, Raidl GR. Balancing bicycle sharing systems: an analysis of path relinking and recombination within a GRASP hybrid. In: International conference on parallel problem solving from nature. Springer; 2014, p. 792–801.
[43] Ren Y, Meng L, Zhao F, Zhang C, Guo H, Tian Y, et al. An improved general variable neighborhood search for a static bike-sharing rebalancing problem considering the depot inventory. Expert Syst Appl 2020;160:113752.
[44] Vergeylen N, Sörensen K, Vansteenwegen P. Large neighborhood search for the bike request scheduling problem. Int Trans Oper Res 2020;27(6):2695–714.

[45] Kabra A, Belavina E, Girotra K. Bike-share systems: Accessibility and availability. Manage Sci 2020;66(9):3803–24.
[46] Raviv T, Tzur M, Forma IA. Static repositioning in a bike-sharing system: models and solution approaches. EURO J Transp Logist 2013;2(3):187–229.
[47] George DK, Xia CH. Fleet-sizing and service availability for a vehicle rental system via closed queueing networks. European J Oper Res 2011;211(1):198–207.
[48] Hulot P. Towards station-level demand prediction for effective rebalancing in bike-sharing systems. (Master's thesis), École Polytechnique de Montréal; 2018.
[49] Forgy EW. Cluster analysis of multivariate data: efficiency versus interpretability of classifications. Biometrics 1965;21:768–9.
[50] Liang J, Jena SD, Lodi A. Dynamic rebalancing optimization for bike-sharing systems: A modeling framework and empirical comparison. European J Oper Res 2024;317(3):875–89.