Research Article

Modelling of User Behaviour for Static Rebalancing of Bike Sharing System: Transfer of Demand from Bike-Shortage Stations to Neighbouring Stations

Roberta Costa Affonso [ID], Florent Couffin, and Patrice Leclaire

Quartz Laboratory (EA 7393), Supméca, 93407 Saint-Ouen, France

Correspondence should be addressed to Roberta Costa Affonso; roberta.costa@supmeca.fr

Bike sharing systems are becoming more and more common around the world. One of the main difficulties is to ensure the availability of bicycles in order to satisfy users. To achieve this objective, managers of these systems set up rebalancing vehicles that displace bicycles to stations that are likely to be in a situation of bike shortage. In order to determine which stations must be supplied on a priority basis and the number of bicycles to be supplied (named in this paper as rebalancing plan), the aim is generally to reduce the lost demand for each station, *i.e.,* the gap between the demand for bicycles and the number of bicycles at a station. On the one hand, this paper proposes an algorithm that evaluates the lost demand in a more realistic way, by describing the behaviour of users faced with a bike-shortage station. It takes into account the possibility that a proportion of users who cannot find bicycles will move to a neighbouring station that is not empty. This proportion depends on the distance between stations and corresponds to the number of users willing to walk a given distance to a neighbouring station. On the other hand, this algorithm provides the value of the objective function to be minimized to a static rebalancing plan algorithm based on a Random Search metaheuristic. The quantities of bicycles to be picked up and dropped off at each station are calculated in a static rebalancing context. The calculation of lost demand based on this algorithm, which simulates user behaviour, was compared with that one obtained by the classical method on a real numerical example obtained from the open data of Parisian Vélib' (more than 1200 stations). In addition, the efficiency of the rebalancing algorithm coupled with the user behaviour simulation algorithm was evaluated on this numerical example and allowed to obtain very good results compared to the rebalancing performed by the system operator.

1. Introduction

Most medium and large cities have installed a bike-sharing system (BSS) since the first appearance of this type of system in Amsterdam in 1965. These systems are part of sustainable development in urban areas. They have experienced a particularly strong growth in the last 15 years [1], with four generations of systems following one another [2]. The study of these systems has given rise to numerous works in the different scientific communities [3] which the main themes are the following: factors & barrier, system optimization, behaviour & impact, safety & health, and sharing economy [4].

Although sometimes managed by private operators, these systems constitute a public transport service [5]. Their objectives are to reduce congestion, gas emissions, and noise and to offer a flexible and cheap means of transport while having a beneficial effect on the health of users [6]. They allow users to make relatively short journeys, of the order of 2.5 km [7] while integrating with other modes of transport in the intermodality context [8]. In addition, these systems provide a flexible mode of transport that improves "first mile/last mile" connections [9]; this distance is considered too long to walk between home and public transport and/or public transport and the workplace [10].

Currently, BSS is divided into two broad categories [11] (not exhaustive, as hybrid systems exist):

(i) The BSSs with dockings, in this case, the stations have dockings (third generation of BSS) to which the bicycles are hooked up.

(ii) The BSSs without dockings (part of the fourth generation of BSS), in this case, the bicycles have an autonomous hook system and are generally geo-localisable. This category is subdivided into two subcategories. The first one concerns systems in which bicycles are grouped together at defined parking areas [12]. The second one concerns systems in which bicycles can be parked in any accessible area of the urban area.

This paper focuses on systems containing stations with dockings, which are the most widespread category, particularly in Europe [13] and worldwide. Indeed, Chen et al. [14] estimate that the fourth generation of BSSs constitutes less than 20% of all systems installed worldwide.

These systems must meet the needs of users to make trips between two stations. A user must therefore find an available bicycle at the departure station and an available docking at the arrival station [15]. Some stations have a tendency to completely empty and others to completely fill up for specific time slots. For example, in the early morning, stations located in residential areas empty out and those located in working areas fill up. This unavailability of bicycles and dockings degrades the quality of service of these systems. For this reason, some operators deploy fleets of rebalancing vehicles that pick up bicycles from full stations and bring them to empty stations [16]. When rebalancing operations occur at night, when the system is not in use, it is named static rebalancing [17]; when they occur while the system is in use, it is named dynamic rebalancing [18]. Other operators implement pricing incentive policies to encourage users to participate in rebalancing [19].

In the context of static or dynamic rebalancing, these operations impact the operating costs of these systems ("*The operational cost of Vélib' for the redistribution a bicycle is about $3*" [20]). They can therefore only be carried out to a limited extent [21]. Thus, from the manager's point of view, it is necessary to identify the stations that need to be rebalanced as a priority.

Our interest in this paper focuses on the problem of static rebalancing of BSS. In the literature, they are often addressed by optimization models whose main objective is either to minimize the bicycle and docking shortage over the entire system or to minimize the amount of resources deployed for rebalancing and/or their use. These studies do not take into account the condition of stations close to the stations in shortage. However, the user may possibly walk a short distance to look for or drop off a bicycle there. For example, Faghih-Imani and Eluru [22] estimate that the arrival and departure rates of a station are related to those of neighbouring stations. The authors show that part of the demand for bicycles (*resp.,* dockings) from a given empty (*resp.,* full) station can be transferred to a neighbouring station that is not empty (*resp.,* not full). Nevertheless, a customer who cannot find a bicycle is usually considered to be leaving the system. Failure to take this user behaviour into account generates errors in the models used to improve the rebalancing of BSS.

Thus, we propose a method that simulates user behaviour to define a static bicycle rebalancing plan in order to reduce the number of lost demands, improving so user satisfaction. This static rebalancing plan contains the stations and quantities of bicycles to be dropped off and the stations and quantities of bicycles to be picked up. A rebalancing plan does not explicitly include the vehicle routes for moving the bicycles.

The proposed method is based on an optimization-simulation coupling. The objective of the optimization algorithm, which determines the static rebalancing plan, is to minimize the number of lost demands (corresponding to a customer who leaves the system without finding a bicycle). The algorithm, which simulates user behaviour, realistically evaluates the number of lost demands. This algorithm accurately represents the behaviour of users in the presence of stations with a bike shortage.

We seek to demonstrate the feasibility of using an algorithm simulating user behaviour to improve the evaluation of lost demand. Then, we use it subsequently to improve the construction of the rebalancing policy applied by the operators of these systems. As an application case, we used the user trip and bicycle availability data from the open data of the Vélib' system at Paris, operated by JCDecaux.

This article is structured as follows: in the following section, we present a synthesis of the state of the art concerning rebalancing, and more specifically the objectives and the solving methods used for the optimization of static rebalancing. We then present our overall approach and principles in Section 3. Algorithms for evaluating lost demand and optimizing static rebalancing are detailed in Section 4. The applications and results are discussed in Section 5. Finally, in Section 6, we summarize the contributions of our work and propose some perspectives.

2. State of the Art

2.1. Generalities. BSS customer satisfaction is highly dependent on the availability of bicycles and dockings at the stations, but also on the level of trip coverage, which measures the proportion of users who find a station available and at an acceptable distance from their points of origin and destination [23]. Short rental times and very high spatial and temporal irregularity of user demand lead to imbalances in the distribution of bicycles [24]. The problem of rebalancing bicycles is therefore one of the main problems in the operation of BSS. According to Fishman and Schepers [25], "*rebalancing refers to bikeshare operators moving bicycles across the network, to maintain a reasonable distribution across docking stations.*" Thus, issues related to rebalancing include defining the stations to be rebalanced, the number of bicycles to be loaded/unloaded, and optimal routes for rebalancing vehicles based on economic requirements [3] such as the number or capacity of rebalancing vehicles, the

number of rebalancing operations, or the number or distances (or duration) of rebalancing tours [26]. The aim of the decision-maker is finally to find the best compromise between economic aspects and customer satisfaction. Both can be integrated as constraints of the problem to be addressed or criteria to be optimised.

Rebalancing is not carried out in the same way depending on the studied time slot of the day, as mentioned in Section 1. Thus, two different types of rebalancing are studied in the literature. The corresponding optimization problem can be classified as Static Bicycle Rebalancing Problem (SBRP) and Dynamic Bicycle Rebalancing Problem (DBRP) as proposed in [27]. SBRP is based on the assumption that the number of bicycles of each station remains the same or changes slightly throughout the rebalancing period, without affecting the outcome of the rebalancing. As such, SBRP is generally conducted at night or when the movement of bicycles does not impact the operation of the system. The dynamic version of this problem assumes that bicycle movements have a significant impact on system user demand, affecting the rebalancing outcome. In this study, we are particularly interested in the SBRP, so we will focus on the work carried out on this topic. This state of the art is not exhaustive. Interested readers can refer to [27, 28], which present a review of studies on this topic.

2.2. Static Bicycle Rebalancing Problem. In the case of static rebalancing, many SBRP formulations have been proposed, differing in the assumptions considered, the objectives to be optimised, and the constraints to be integrated. Most of the work is concerned with determining the vehicle routes for rebalancing operations. In the case where the system is partitioned into small zones, only one vehicle is in charge of rebalancing.

Chemla et al. [29] consider that a vehicle can visit a station several times, allowing temporary storage of bicycles at a station. The authors formulate the problem as a special case of a Pickup and Delivery Problem (PDP) with a single vehicle and a single type of bicycle. The authors propose a mathematical formulation and a relaxation of the problem. It is solved by a branch and bound and provides a good lower bound of the optimal solution. This optimal solution is then used as an initial solution of a Taboo Search. The latter is then able to obtain a solution with a cost similar to the optimal solution.

Erdoğan et al. [30] study vehicle routing for static rebalancing. The objective is to minimize the overall cost of the route (assumed to integrate the consumption and duration of the route, and CO_2 emission of the rebalancing vehicle). It corresponds to a set of stations to be visited from a depot, ensuring that the vehicle leaves and arrives at the depot to satisfy the demand for bicycles from the initial number of bicycles at each station. The problem is defined as a linear integer program and solved by a heuristic to obtain an exact solution.

Cruz et al. [31] consider the same problem and propose a hybrid Iterated Local Search (ILS) combined with a Randomized Variable Neighborhood Descent (RVND). Their method allows finding the optimal solution for a good number of instances or even to improve the best known solutions.

Papazek et al. [32] and Rainer-Harbach et al. [33] address the routing of several rebalancing vehicles. The objective is to minimize the absolute deviation between target and final fill levels, and the number of deposit/removal operations and the duration of the routes. The authors consider the capacity constraints of vehicles and rebalancing stations. The target fill level is defined by a statistical demand forecasting model such as [34]. The problem is modelled by an oriented graph representing the network. The nodes represent the stations, and the arcs are associated with the travel times of the vehicles. The end result is the vehicle route. Papazek et al. [32] solve the problem using the Greedy Randomized Adaptive Search Procedure (GRASP) metaheuristic, while Rainer-Harbach et al. [33] use a Variable Neighborhood Search (VNS) metaheuristic. Rainer-Harbach et al. [35] study the same problem from the same formulation using the PILOT (Preferred Iterative Look-Ahead Technique) method for the construction of initial solutions for the VND (Variable Neighborhood Descent) and GRASP metaheuristics. Some works integrate other operational functionalities, for example, [36], which adds the collection of bicycles to be repaired. Gaspero et al. [37] address the same problem, but each route must be completed within a limited time frame. Constraint programming is used to model the problem formulated as a routing problem.

Forma et al. [38] adapt a cluster-first route-second approach to subdivide the problem by vehicle-associated zones. With the same idea of dividing the network into different zones, Lv et al. [39] propose a hybrid algorithm for BRP. A destroy and repair algorithm is developed to improve the clusters, and an adaptive variable neighborhood search algorithm is designed to conduct intracluster and intercluster vehicle routing optimization. The objective function consists of traveling cost and inventory cost.

Dell'Amico et al. [40] develop a destroy and repair metaheuristic for the BRP, using a set of techniques based on properties of feasible paths altered by a neighborhood operator, in order to speed up the computational effort.

In most cases, the route(s) of the rebalancing vehicles is(are) explicitly indicated by the succession of stations visited and the number of bicycles picked up or dropped off at each of them.

We note that the majority of the work dealing with SBRP is aimed at improving the efficiency of rebalancing according to the criteria of cost, distance or duration of rebalancing, or maximizing user satisfaction. This can be defined as the difference between the number of bicycles obtained after rebalancing and user demand, as in [32, 33] or by a penalty function proposed by Kaspi et al. [41] or as a cumulative duration during which stations are not in a state of equilibrium, as proposed by Kadri et al. [42].

In this article, we adopt another criterion measuring the effectiveness of rebalancing. We aim at minimizing lost user demand, considering a limited number of bicycles to be

redistributed. The goal here is to propose a rebalancing plan that minimizes lost demand, and according to this plan, the manager can then organize the rebalancing vehicle routes.

The main originality of our method is to consider that users can walk to a nearby station to find a bicycle, when the desired starting station is empty. Indeed, this user behaviour is not integrated in the work in the field of SBPR. The existence of such behaviour has been emphasized and taken into account by some authors, in other fields of application. Farghih-Imani and Eluru [22] and Rudloff and Lackner [34] consider this behaviour for the development of a BSS demand forecasting model, without being subsequently applied to a system rebalancing approach. These authors consider in their model an increase in demand from neighbouring stations when a station is empty. It is therefore important to take this phenomenon into account when defining the priority stations to supply bicycles during a rebalancing operation. It is probably more judicious to supply an empty station when its neighbouring stations are also empty, than to supply an empty station when the neighbouring stations have a stock of bicycles.

Chiariotti et al. [43] take into account the possibility of a user to walk to a neighbouring station, but in the context of dynamic regulation. The authors develop a rebalancing method associated with a method of monetary incentives for users. The users are encouraged to walk until they reach a congestion-free station instead of just taking the first bike they find. However, this method includes user displacement cost, whereas some users can displace without monetary incentive.

In our study, we consider that users may walk, without any monetary incentive, to a neighbouring stations if they do not find a bicycle at a desired station. Our goal is to define an algorithm that evaluates the lost demand of each station in a realistic way by taking into account the availability of its neighbouring stations and simulating user behaviour. We will compare on a real system this lost demand calculation algorithm with the classical method. It will allow us to demonstrate that both approaches give significantly different results. Also, we couple this first algorithm to an optimization algorithm for establishing the rebalancing plan in terms of bikes to be picked up or dropped off at stations. Thus, we will use the first algorithm for evaluating the objective function to be minimized. The goal is to determine a rebalancing plan to demonstrate the relevance and feasibility of our approach on a real system application.

3. Overall Approach and Principles

In this section, we will present the proposed approach, which aims to propose a static rebalancing plan that is efficient in terms of lost user demand for a BSS. Then, we will explain the principle on which the method is based. This principle is to integrate the *behaviour of users* who could move from an empty station where they wanted to take a bicycle and go to a neighbouring nonempty station to get one. The users who do not finally find bicycles represent the lost (user) demand. In order to describe this principle, we must first introduce the notion of acceptance rate. It is related to the number of users

willing to go and fetch a bicycle from one of the neighbouring stations. Finally, we will present how we estimate the user demand for bicycles at a station during the time that this one is empty.

3.1. Global Approach. The proposed method implements a coupling of an approximate optimization algorithm (metaheuristics) with an algorithm that simulates user behaviour to compute lost demand. This is a simulation optimization approach, as depicted in [44]. A solution is a rebalancing plan and is characterized by the number of bikes picked up from a station or dropped off to it. The initial stock of these stations will be modified according to this rebalancing plan. Then, the simulation algorithm acts as a function that evaluates the lost demand of a solution. This lost demand is then used as the criterion to be minimized by the optimization algorithm. Therefore, for each solution explored by the optimization algorithm, the associated lost demand is evaluated by the simulation algorithm.

As soon as a station is empty and there is still demand associated with this station, a part of this demand is transferred to the nearest nonempty station. Other distribution rules of users to one or more neighbouring nonempty stations are possible, for example, "the station with the most bicycles available" or "the nearest station in the direction of the user's journey." The transferred demand is based on an acceptance rate that corresponds to the proportion of users who are willing to walk a certain distance to reach a neighbouring station.

The optimization and simulation algorithms are detailed in Section 4.

3.2. Acceptance Rate. As mentioned before, the demand transferred from one station to another, when the first one is empty, is relative to the rate of users who may walk to reach a neighbouring station that is not empty [45]. To statistically assess the maximum distance users are willing to walk, we carried out a survey of 136 Vélib' system users. The selected users are occasional users, who take a bicycle at least once a month on a working day, and regular users, who take a bicycle every working day. The question asked was "What is the maximum distance you may walk to get a bicycle from an empty Vélib' station?" The results are summarized in Table 1. For example, 86 out of 136 users (63%) agree to walk till 300 meters to get a bicycle.

We can note through Table 1 that only 8% of users agree to walk beyond this distance. To avoid associating a station with a large area, the entire neighbouring area is limited to 500 meters around a studied station. The number of users who agree to walk more than 500 meters to get a bicycle is neglected. Also, other studies of the literature [22, 46] mention that the maximum distance a user is willing to walk to reach a nonempty station is 500 meters.

The 500-meterzone around the station i is named *acceptance zone* Figure 1. This acceptance zone is broken down into concentric circles every 100 meters. The areas between two consecutive circles are referred to as *neighbouring zones,* Zones 1 to 5 in Figure 1, and may contain stations, neighbouring of Station i.

TABLE 1: Survey results concerning the distance users are willing to walk.

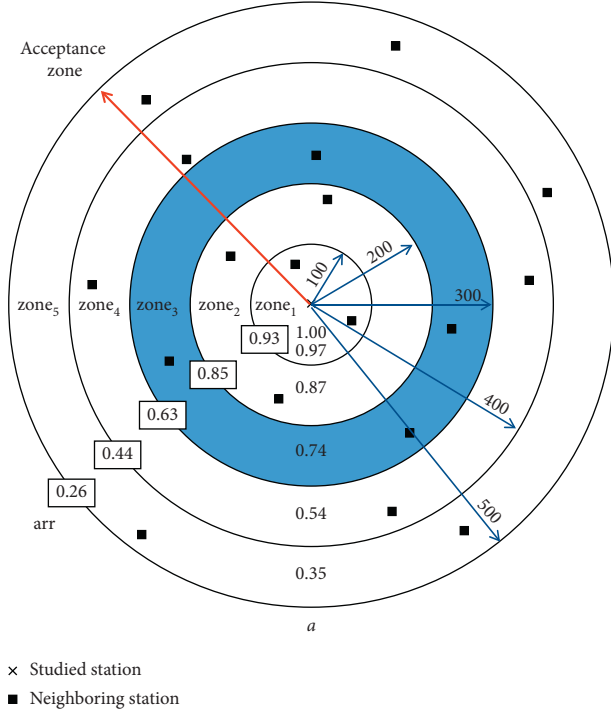| Distance (m)	Number of users	User proportions
0	136	1.00
100	127	0.93
200	115	0.85
300	86	0.63
400	60	0.44
500	35	0.26
More than 500	11	0.08

× Studied station

■ Neighboring station

FIGURE 1: Acceptance and neighbouring zones of a station i.

TABLE 2: Illustration of user behaviour for a three-station system.

	Station 1	Station 2	Station 3
Initial system status			
Initial stock	10	30	20
Initial demand	20	10	30
Distribution of users after simulation of user behaviour			
Residual demand	10	0	10
Residual stock	0	20	0
Transferred demand	7	0	5
Backlog demand	0	12	0
Residual stock	0	8	0
Lost demand	3	0	5

We define the acceptance rate ($a_{i,j}$) of a neighbouring Zone j as the average of the proportion of users ($\mathrm{arr}_{i,j}$) agreeing to walk at a distance between the radius of the inner circle ($\mathrm{arr}_{i,j-1}$) and the radius of the outer circle ($\mathrm{arr}_{i,j}$), or

$$a_{i,j} = \frac{\mathrm{arr}_{i,j-1} + \mathrm{arr}_{i,j}}{2}. \tag{1}$$

For each neighbouring zone, Figure 1 presents its radius, the proportion of users agreeing to walk to the outer limit of this zone, and its acceptance rate. For example, Zone 3 is located between two circles, which respective radii are 200 and 300 meters; 63% of users agree to pick up a bicycle to the outer limit of this zone, and the acceptance rate for this zone is equal to 74%.

3.3. User Behaviour. We will explain here how customer behaviour is integrated into the lost demand evaluation algorithm, which will be detailed in Section 4.

To understand user behaviour, we illustrate it through a case of a system consisting of three stations. It integrates the possibility that users who cannot find bicycles can go and fetch one from a station within the acceptance zone (less than 500 meters away).

Table 2 describes the state of the network and its evolution:

(i) The initial stock corresponds to the stock of bicycles of each station before the first users arrive

(ii) The initial demand corresponds to the number of users wanting to take a bicycle at each station when the system starts up

Once users had removed the bicycles corresponding to the initial demand at each station, a 20-bicycle demand, named *residual demand*, could not be met (10 at Station 1 and 10 at Station 3). A residual stock of 20 bicycles is present at Station 2. In a conventional approach, these 20 residual demands would be considered as permanently lost and the 20 dissatisfied users would leave the system.

If Station 2 is close to Stations 1 and 3, some of the dissatisfied users will go there to borrow a bicycle. Let us assume that Station 2 is located 250 meters (Zone 3) from Station 1 and 350 meters (Zone 4) from Station 3. In this case, 74% of dissatisfied users agree to walk from Station 1 to Station 2 and 54% from Station 3 to Station 2.

Multiplying the residual demand by these acceptance rates (and rounding the real numbers to default integers) would result in 7 users moving from Station 1 to Station 2 and 5 moving from Station 3 to Station 2. We refer to this as the *demand transferred* from the source station to the nearest neighbouring station. The sum of the demand transferred to the same neighbouring station is called *backlog demand*. In our example, the backlog demand of Station 2 is 12 bicycles. Considering the number of bicycles remaining in Station 2, this backlog demand will be satisfied. From now on, the lost demand is only 8 users instead of 20.

If the residual stock of Station 2 was only 8 bicycles instead of 20, 4 users would not have been able to take bicycles. They would then have gone to the first nonempty neighbouring station (applying the corresponding acceptance rate).

3.4. Correction of the Demand. The actual demand for bicycles is equal to the number of bicycles leaving minus the number of bicycles arriving to a station. However, if the

station is empty for a period of time, this evaluation leads to an underestimation of the demand for bicycles. Inaccurate estimates of demand can lead to suboptimal decisions, particularly in terms of rebalancing [47]. Demand must therefore be corrected when a station is out of bicycle. Albinski et al. [48] estimate the actual demand of a station that is out of bicycle for a time slot as the average of the demand on all days under study, corresponding to the same day of the week and the same time slot where the station is not empty. In this paper, a different approach was considered in order to integrate the movement of users to the nearest nonempty station in accordance with the principle described in the previous section.

If a station has no more bicycles for a certain period of time during the time slot under consideration, we take into account the number of requests per minute during the time that this station was not empty. This amount is then multiplied by the number of minutes the station was empty. This value is added at the request of the station. Thus, the nearest nonempty neighbouring station will have its request reduced by this value multiplied by its acceptance rate (in relation to its neighbourhood area). This last correction avoids overevaluating the demand over the whole system for the period under consideration.

For example, if a station had a demand for 15 bicycles from 07:00 to 09:00 am (period studied) but was empty for 30 minutes, we consider its total demand to be 20 bicycles. Also, if the nearest nonempty neighbouring station had a demand for 12 bicycles and is located in Zone 3 (74% acceptance rate), we consider its total demand to be 8.3 bicycles.

4. Proposed Method: Coupling Algorithms for Optimizing Rebalancing Plan and Evaluation of Lost Demand

4.1. Description of the Study Framework. In this section, we will present the notations used, the modelling and the algorithmic principles for evaluating lost demand and optimizing rebalancing plan. The aim of these frameworks is twofold:

(i) Simulating the behaviour of users and the distribution of demand over neighbouring stations, in order to assess the impact of rebalancing on lost demand

(ii) Proposing a rebalancing plan that takes this behaviour into account in order to minimize lost demand, and considering constraints related to the total number of bicycles that can be displaced and to the stations from which bicycles can be picked up for rebalancing operations

4.2. Evaluation of the Lost Demand Taking into Account the Behaviour of Users. We propose to define a lost demand calculation algorithm to integrate the behaviour of users who move to neighbouring stations if they cannot find

bicycles at a desired source station. This algorithm follows the principles described in the User Behaviour section.

Let us assume that we have a set of stations $I = \{1, \ldots, n\}$ of n stations. For each station, we know the neighbouring stations (located in its acceptance zone). This is transcribed by the $n \times n$ matrix $L = (l_{ij})_{i \in I, j \in I}$ containing all interstation distances and the $n \times n$ matrix $A = (a_{ij})_{i \in I, j \in I}$ containing the acceptance rates. Each component a_{ij} provides the user acceptance rate to move from a source station i to a nearby station j. Each station is characterized by its initial stock of bicycles b_i, its capacity c_i, and its initial demand for bicycles d_i.

At each iteration of the algorithm customers take bicycles, if there are enough of them. Some of the customers who have not found a bicycle at a given source station move to the nearest nonempty target station, and the others abandon the system.

The data needed for the algorithm and their notations are as follows:

(i) n: total number of stations in the network.

(ii) $I = \{1, \ldots, n\}$: set of station indices.

(iii) $i(j)$: index of a station, with $i \in I$, $j \in I$.

(iv) k: index of iteration, with $k \in \mathbb{N}$.

(v) b_i: quantity (or stock) of bicycles available at a station $i \in I$ at the beginning of the simulation.

(vi) b_i^k: quantity (or stock) of bicycles available at a station $i \in I$ at iteration k with $b_i^k \geq 0$.

(vii) d_i: initial demand for bicycles of a station $i \in I$.

(viii) d_i^k: backlog demand for bicycles of a station $i \in I$ at iteration k.

(ix) r_i^k: residual demand for bicycles of a station $i \in I$ at iteration k with $r_i^k \geq 0$. This is the backlog demand after taking bicycles from stock b_i^k of the station.

(x) a_{ij}: acceptance rate from a station $i \in I$ to a station $j \in I$ with $0 \leq a_{ij} \leq 1$.

(xi) l_{ij}: distance from a station $i \in I$ to a station $j \in I$ with $l_{ij} \geq 0$.

The studied criteria and the corresponding notations are as follows:

(i) ω_i^k: lost demand of station $i \in I$ at iteration k.

(ii) Ω: total demand lost at iteration k.

Algorithm 1 presents the evaluation of lost demand after dispatching users, whose demand cannot be satisfied at a given source station, to one of the neighbouring stations within the acceptance zone (less than 500 meters).

The backlog demand of each station i is initialized for $k = 0$ to its initial demand $(d_i^0 \longleftarrow d_i)$ and the stock of bicycles to the number of bicycles initially present at the station $(b_i^0 \longleftarrow b_i)$.

At each iteration $k \geq 1$, the backlog demand d_i^{k-1}, from the previous iteration $(k-1)$, of a source station $i \in I$ is processed, in part, by the source station itself according to the number of bicycles present at this station (the number of bicycles b_i^k is then updated: $b_i^k \longleftarrow \max(0, b_i^{k-1} - d_i^{k-1})$). The

Input: The initial station status ($\forall i \in I$: d_i, b_i), the interstation acceptance rate matrix ($A = (a_{ij})_{i \in I, j \in I}$), the interstation distance matrix ($L = (l_{ij})_{i \in I, j \in I}$)
Output: Total lost demand Ω
Begin
   $k \longleftarrow 0$
   $\Omega \longleftarrow 0$
     For all station $i \in I$ **do**
      $d_i^0 \longleftarrow d_i$
      $b_i^0 \longleftarrow b_i$
     End for
     While $\sum_{i \in I} d_i^k > 0$ and $\sum_{i \in I} b_i^k > 0$ **do**
      $k \longleftarrow k + 1$
      //Calculation of residual stock (b_i^k) of stations
      $\forall i \in I, b_i^k \longleftarrow \max(0, b_i^{k-1} - d_i^{k-1})$
      //Calculation of residual demand (r_i^k) of stations
      $\forall i \in I, r_i^k \longleftarrow \max(0, d_i^{k-1} - b_i^{k-1})$
      //Calculation of (i) backlog demand (d_j^k) of the target stations j after transferring users from source stations i which have a residual demand (r_i^k) not null, (ii) lost demand of each station (ω_i^k), and (iii) global lost demand (Ω)
        For all station $i \in I$ **do**
         $d_i^k \longleftarrow 0$
        End for
        For all station $i \in I$ such as $r_i^k > 0$ **do**
         $\omega_i^k \longleftarrow 0$
         Let be j a nonempty station (such as $b_j^k > 0$), which is acceptable (such as $0 < a_{ij} < 1$) and the closest, in the sense of the matrix L (if there are several stations at the same distance, the smallest index station is chosen).
         If j exists **then**
           $d_j^k \longleftarrow d_j^k + r_i^k a_{ij}$
           $\omega_i^k \longleftarrow r_i^k (1 - a_{ij})$
           $\Omega \longleftarrow \Omega + \omega_i^k$
         End If
        End For
     End While
End

ALGORITHM 1: Algorithm for lost demand evaluation considering user behaviour.

remaining backlog demand then becomes the residual demand, $r_i^k \longleftarrow \max(0, d_i^{k-1} - b_i^{k-1})$ of this source station i.

Part of the residual demand of the source station i is moved to a neighbouring station, named *target station j*. This station j is selected as the closest one in the acceptance zone, using the interstation distances contained in the matrix L, which respects the following conditions: nonempty station ($b_j^k > 0$), with a not null acceptance rate a_{ij} of the source station i toward the target station j. The backlog demand of the target station j is updated as follows: $d_j^k \longleftarrow d_j^k + r_i^k a_{ij}$. The term $r_i^k a_{ij}$ is the demand transferred from the station i to the station j. The remainder of the residual demand is considered as lost demand: $\omega_i^k \longleftarrow r_i^k (1 - a_{ij})$. The algorithm stops when, for all stations, there is no more residual demand to be dispatched.

4.3. Coupling between a Random Search and the Lost Demand Evaluation Algorithm.

The optimization problem studied consists in determining a rebalancing plan, by characterizing the number of bicycles picked up or dropped off during rebalancing operation. The objective function is to minimize the number of lost bicycle demand, evaluated by Algorithm 1. The constraints considered are as follows:

(C1) Obligation to pick-up bicycles only at certain stations, named suppliers' stations, defined by the operator

(C2) Limitation of the total number of bicycles that can be displaced by rebalancing operations, also defined by the operator

(C3) Keeping the total number of bicycles, during rebalancing operations, in the system

(C4) Respecting station capacities (in number of bicycles)

We have chosen to use an approximated optimization scheme in the form of a single-solution-based metaheuristic, as named in [49]. This will allow disturbing a solution (a rebalancing plan) in order to obtain a better solution in the sense of the studied criterion (the total number of lost demand). The solutions will be evaluated, at each iteration, by using Algorithm 1.

The aim here is to show the feasibility of such an optimization approach coupled with the proposed user behaviour simulation model. The metaheuristic, more precisely a Random Search, is therefore deliberately relatively simple; we will see in Section 5 that results obtained are very satisfactory, and we will discuss possible improvements.

To set up this metaheuristic, we have chosen to represent a solution (rebalancing plan) by a set indexed on the stations. It contains, for each station, the number of bicycles picked up or dropped off to this station. This set is noted $x = (x_i)_{i \in I}$ where, for each station $i \in I$, x_i is the number of bicycles picked up (negative number) or dropped off (positive number) at this station.

The additional data and the corresponding notations are as follows:

(i) b_i: the number of bicycles initially present at a station $i \in I$.

(ii) x_i: the number of bicycles picked up or dropped off at a station $i \in I$ by rebalancing operations, from the current solution x.

(iii) c_i: capacity in dockings (or maximum number of bicycles) of a station $i \in I$ with $c_i > 0$.

(iv) p_i: which is 1 if the station $i \in I$ is a picking station, 0 otherwise.

(v) Δ: total number of bicycles displaced by rebalancing operations (*i.e.*, total number of bicycles picked up or total number of bicycles dropped off by rebalancing operations).

(vi) Δ_{max}: maximum number of bicycles that can be displaced by rebalancing operations; Δ_{max} is a parameter.

(vii) k_{max}: maximum number of iterations of metaheuristics; k_{max} is a parameter.

(viii) Ω: total number of lost demand associated with the solution x evaluated by using Algorithm 1.

(ix) δ: the number of bicycles displaced by a neighbourhood operator application.

In Algorithm 2, we define a neighbourhood operator to modify the number of bicycles present at each station, as a rebalancing operation would do, *i.e.*, by picking up bicycles from one station and dropping them off at another station. The neighbourhood operator randomly selects two stations i_1 and i_2 so that bicycles can be picked up in i_1 (*i.e.*, $b_{i_1} + x_{i_1} > 0$) and dropped off to i_2, while respecting the capacity constraints of the station i_2 (*i.e.*, $b_{i_2} + x_{i_2} < c_{i_2}$). The number δ which represents the number of bicycles displaced from i_1 to i_2 is randomly chosen between 1 and the maximum number of bicycles that can be displaced between these stations, *i.e.*, $\min(b_{i_1} + x_{i_1}, c_{i_2} - (b_{i_2} + x_{i_2}))$.

Each new solution is evaluated using Algorithm 1, so as to obtain a value of the total lost demand Ω which is the objective function, to be minimized, of the metaheuristic. If the lost demand Ω' of a neighbouring solution $x\prime$ is less than the lost demand Ω of the current solution x, then the current solution x is replaced by the neighbouring solution $x\prime$.

We note that the first constraint, noted C1, imposes that bicycles can be picked up only from the suppliers' station, where each station $i \in I$ is characterized by Boolean p_i which is 1 if the station is *enabled for a bicycle picking operation* and 0 otherwise. The second constraint, noted C2, consists in limiting the total number Δ of moveable bicycles, by a

maximum quantity noted Δ_{max}. This constraint aims to consider a realistic number of moveable bicycles during the rebalancing operation.

The C3 constraint is respected by construction by the expressions $x'_{i_1} \longleftarrow x_{i_1} - \delta$ and $x'_{i_2} \longleftarrow x_{i_2} - \delta$ of the neighbourhood operator.

5. Experiments and Results

In order to show the relevance of the proposed method, we have carried out three different experiments in the Vélib' system at Paris. These three experiments have different objectives. The first one aims at showing and analysing the gap in the evaluation of the loss of demand between the proposed method and the method more frequently used in the literature (detailed below). The second application aims to show the interest of using this method to define the static rebalancing plan of the Paris Vélib' system, by applying the coupling defined in Section 4. The estimated lost demand after implementing this plan will be compared to the estimated lost demand after implementing the rebalancing solution carried out by JCDecaux (operator of this system). Finally, we will apply the method in a prospective framework, in which we will evaluate the static rebalancing plan by considering a progressive increase in demand. This final test stage is intended to show that the method can also be useful for studying a system that is in an evolutionary phase in terms of demand growth. Before presenting these experiments, we will introduce the main characteristics of the Vélib' system, the data used and the assumptions considered.

5.1. Study Case: Vélib' System. The Vélib' system was launched by JCDecaux in 2007 and encompassed around 17,000 bicycles and 1,230 docking stations covering Paris and suburban areas. It offered nonstop service (24/7), and each station was equipped with an automatic rental terminal. By July 2014, there have been more than 200 million trips and more than 274,000 annual subscriptions.

The data used in this study were provided by JCDecaux. They concern the flow of bicycles entering and leaving each station of the network, and the status of the stations (number of bicycles and dockings available at each station). In order to obtain data corresponding to a homogeneous system behaviour, we have only used the one of the worked days (Saturday and Sunday excluded) for the period from September 04th, 2017 to October 13th, 2017. This period was chosen because it does not include school holidays or work-free days. In fact, the users' demand is very different between the working days and the week-end days and holidays.

We consider that static rebalancing is carried out between 00:00 and 06:59 am, when user movements are considered negligible. Then, we study the starting period of the system (from 07:00 to 09:00 am) until dynamic rebalancing begins.

The number of bicycles initially present at each station (initial stock), for a given date, corresponds to the number of bicycles actually present at these stations at 00:00 am, from

which bicycles are picked-up or dropped off according to the rebalancing plan.

The stations, from which bicycles are picked up for static rebalancing, are the same as those frequently used (in terms of frequency and quantity of bicycles picked up) by JCDecaux in its rebalancing plan.

For the first two experiments, we used the demand of each day of the study period during the morning peak hours (from 07:00 to 09:00 am). In 2017, this demand corresponds to an average of 6200 trips, and the standard deviation is evaluated at 995.

We would like to point out that the demand for one of the days, included in the studied period, was abnormally low (1527 trips). We do not know exactly the reason for this drop in demand, so we did not consider this day in this study, thus it includes 29 working days.

The three experiments of the method were carried out for 100,000 iterations of Algorithm 2. After the presentation of the experiments, we analyse the convergence of the method execution.

5.2. Experiment 1: Analysis of Lost Demand Evaluation Method.
The main indicator we use to evaluate and analyse the results of the proposed method is lost demand, which largely reflects client dissatisfaction. Thus, we need to verify the value of such an indicator. We measure and analyse the difference in the lost demand estimation when we integrate user behaviour (as described in the User Behaviour section) and when we do not integrate it in the estimation. We will call here the first case of LDUB, for Lost Demand considering User Behaviour, and the second case of LDC, for Lost Demand considering Classic evaluation. The latter calculation method considers that if the user does not find a bicycle in the desired station, he leaves the system immediately. It is therefore counted as lost demand.

We applied both methods of calculation for the 29 studied days. For each day, we considered the actual stock of each station in the system at 07:00 am. To estimate the LDUB, we used the Algorithm 1 proposed in Section 4, and for the case of the LDC, we calculated the difference between the demand and the initial inventory. If the value is positive, it is counted as lost demand. Figure 2 shows the graph of lost demand for both calculation methods. We can see that the proposed evaluation method estimates a much smaller loss of user demand, representing a gap of 36% on average, with a small standard deviation of 5%. This therefore shows a significant difference in the evaluation of the loss demand. It should be noted that the correction of the demand is estimated at 4.5% of the average demand of the 29 days studied (6200 bicycles).

Thus, decision-makers, who rely on the classic indicator, will probably take less efficient decisions. It will lead them to overestimate by an average of 23% (the difference between LDUB and LDC in relation to the number of bicycles displaced) the number of bicycles to be displaced during static rebalancing operations Figure 3.

5.3. Experiment 2: Static Rebalancing of the Vélib' System.
We compared the results, in terms of LDUB, of the JCDecaux rebalancing plan and the results of the proposed method for the 29 days under review.

In order to evaluate JCDecaux's rebalancing plan, we considered the number of bicycles in each station of the system at 00:00 am (beginning of the shutdown period or beginning of the static rebalancing of the system). Using JCDecaux's open data, we have registered the number of bicycles dropped off and picked-up per station, as part of a rebalancing operation, for each day between 00:00 and 07:00 am (rebalancing period). We then used Algorithm 1 to estimate the lost demand (LDUB). The results are shown in Figures 4(a) and 4(b). After the rebalancing carried out by JCDecaux, the system shows a demand loss estimated at 230 users on average. We can see that the loss of demand on day 5 is much higher than on the other days studied. If we do not consider day 5 in our analysis, the demand loss for JCDecaux is estimated at 214 users on average (standard deviation 38.6), while the demand loss for the proposed method is estimated at only 39 users on average (standard deviation 38.7). This represents an improvement of approximately 82% in the number of lost demands. Furthermore, if we compare the number of bicycles to be displaced for each rebalancing plan Figure 4(b), there is a significant difference of 662 bicycles between the two rebalancing frameworks. During the 29 studied days, JCDecaux rebalanced an average of 1,928 bicycles (standard deviation of 184.4), and this difference in the number of bicycles rebalanced represents 34% of the bicycles rebalanced by JCDecaux. We believe that this difference is a good indication that the resulting rebalancing plan may lead to less costly solutions for rebalancing vehicle routing than those carried out by the system operator. Especially as the stations from which bicycles are picked up for static rebalancing are the same as those frequently used by JCDecaux.

5.4. Experiment 3: Forward-Looking Framework for Incremental Demand Growth.
In order to show an example of an experiment in which the proposed method can be used, a theoretical scenario has been defined based on the example of the Vélib' system. The aim here is to show the usefulness of a method that takes user behaviour into account, to evaluate prospective scenarios. This may concern a system that is not yet stabilized in terms of demand, for example, at the beginning of deployment. This context is represented here by a progressive growth in demand.

To do this, we have considered that all the bicycles in the system (here about 16,500 bicycles) are distributed equitably and proportionally to the capacity of each station. For this theoretical study, it is assumed that overall demand follows a normal distribution. We simulated the demand by varying the average between 1000 and 15,000 bicycles. This set of normal distribution was approximated by applying a proportionality relationship based on the mean and standard deviation of the demand of the real system over the 29 days

Input: Station capacity ($\forall i \in I$: c_i), the initial station status ($\forall i \in I$: d_i, b_i), the interstation acceptance rate matrix ($A = (a_{ij})_{i \in I, j \in I}$), the interstation distance matrix ($L = (l_{ij})_{i \in I, j \in I}$), the maximum number of movable bicycles Δ_{max}, the list of supplier stations (*i.e.* $p_i, \forall i \in I$), and the number k_{max} of maximum iterations.

Output: The rebalancing plan (*i.e.*, the number of bicycles picked up or dropped off, $x_i, \forall i \in I$), the total lost demand Ω evaluated by using Algorithm 1, and the total number Δ of displaced bicycles

Begin

    $k \longleftarrow 0$ //iteration counter

    $\Omega \longleftarrow +\infty$ //total loss of the current solution

    $\Delta \longleftarrow 0$ //total number of bicycles displaced

    For all $i \in I$ **do**

    $x_i \longleftarrow 0$ //initialization of the solution variables (or by a constructive heuristic)

    End for

    While $k \leq k_{max}$ **do**

      //Neighbourhood operator

      Randomly select a picking station i_1 ($p_{i_1} = 1$ for compliance with constraint C1) which still has bicycles to be picked up $(b_{i_1} + x_{i_1} > 0)$

      Randomly select a station i_2, different from i_1, such that there is still available dockings for bicycles ($b_{i_2} + x_{i_2} < c_{i_2}$ for compliance with constraint C4)

      Choose a random number δ of bicycles to be displaced from the station i_1 to the station i_2 within the interval $[1, \min(b_{i_1} + x_{i_1}, c_{i_2} - (b_{i_2} + x_{i_2}))]$

      $x_{i_1}' \longleftarrow x_{i_1} - \delta$

      $x_{i_2}' \longleftarrow x_{i_2} + \delta$

      //Checking the constraint of the total number of bicycles displaced

      Calculate the number of bicycles Δ' displaced by solution x'

      If $\Delta' \leq \Delta_{max}$ **then** //compliance with constraint C2

        //Evaluation of the loss Ω' of the neighbouring solution x'

        $\Omega' \longleftarrow$ simulation $(c_i, d_i, b_i + x_i', A, L)$ //call of Algorithm 1

        If $\Omega' \leq \Omega$ **then**

          $x_{i_1} \longleftarrow x_{i_1}'$

          $x_{i_2} \longleftarrow x_{i_2}'$

          $\Delta \longleftarrow \Delta'$ //update of the number of bicycles displaced

          $\Omega \longleftarrow \Omega'$

        End if

      End if

      $k \longleftarrow k + 1$

    End while

End

ALGORITHM 2: Optimization algorithm by metaheuristic (Random Search) using Algorithm 1 for lost demand evaluation.
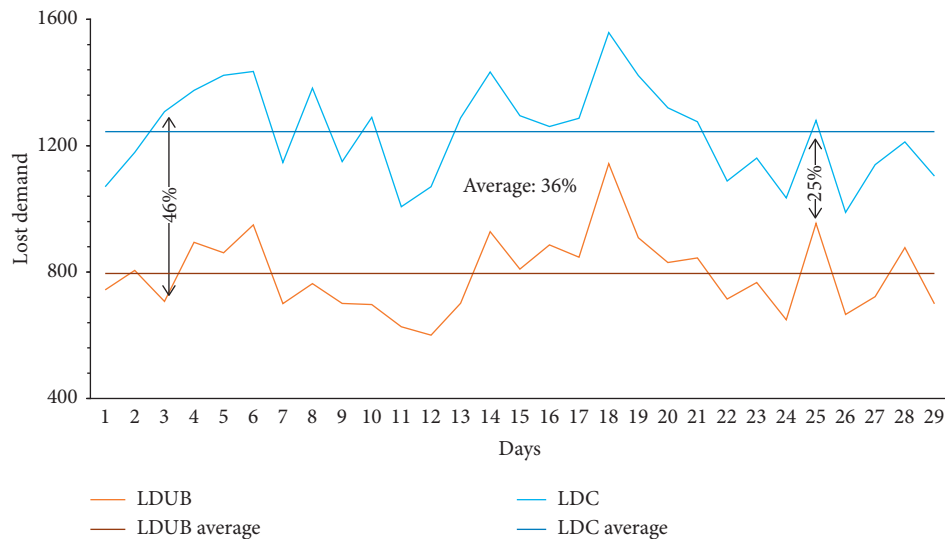

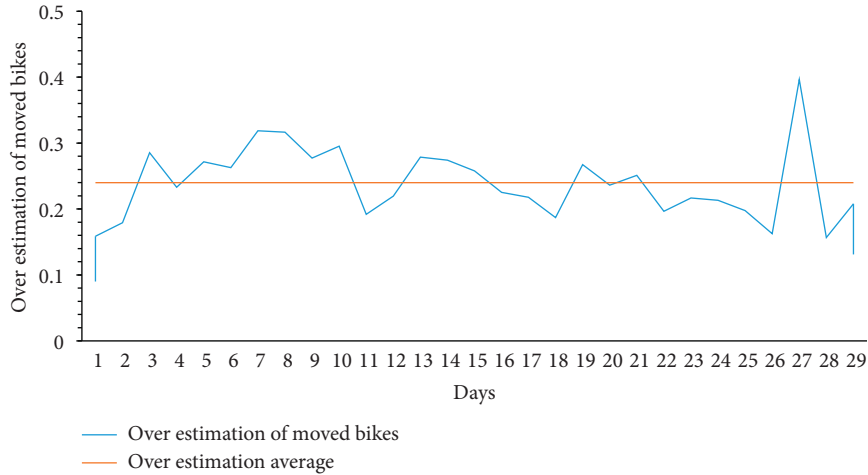FIGURE 2: Difference in lost demand assessment between LDUB and LDC calculation methods.

FIGURE 3: Evaluation of the overestimation of bicycles displaced by rebalancing per day.
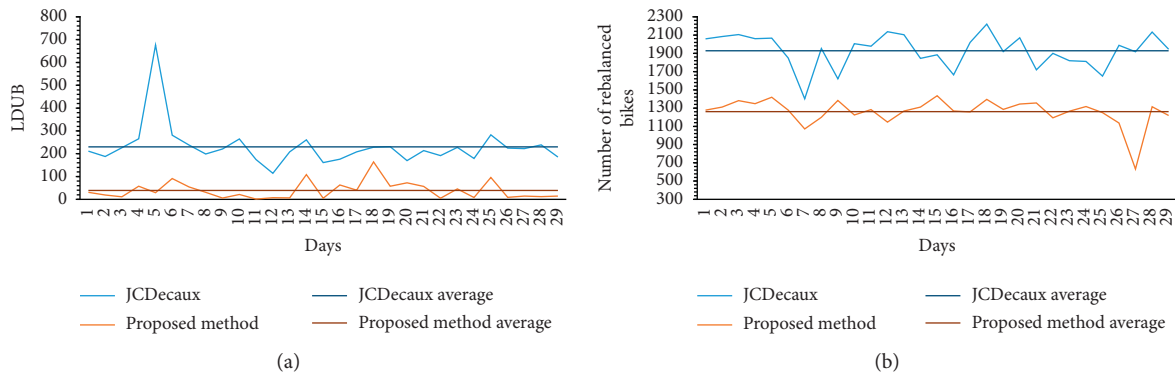

(a)

(b)

FIGURE 4: (a) Estimated LDUB after rebalancing according to the proposed method and according to the operator. (b) Number of bicycles displaced according to the rebalancing of the proposed method and according to the operator's rebalancing.

considered, *i.e.,m* = 6217 and *s* = 1336. For each of the 15 values, we have performed 5 simulation runs. The result we present is the average lost demand obtained for these 5 simulation runs (with a maximum variation coefficient of 5%) for each value of the demand.

In addition, it is considered that the number of bicycles that can be displaced as part of the rebalancing operation is no longer limited. We compared two scenarios: in the first one, we did not consider rebalancing operations. We thus used Algorithm 1 to estimate the LDUB. In the second scenario, we used the proposed method to define the rebalancing plan and then estimate the LDUB.

Figure 5(a) shows the difference in the evolution of the LDUB for the two scenarios. The result of the proposed method is significantly better than the second scenario. Although this result is expected, it is important to note that the LDUB in the second scenario does not exceed 3% of the total system demand.

We also analyse the number of bicycles displaced moved to achieve this performance. Figure 5(b) shows the curve of the ratio of the number of rebalanced bicycles to total demand. We note that for a demand of more than 9000 bicycles, the ratio increases significantly but does not exceed

36% regardless of the value of the considered. The actual JCDecaux system has a ratio of approximately 33%. We can therefore observe that even considering a very strong increase in demand, the proposed method shows promising results.

5.5. Analysis of Method Parameters

5.5.1. Convergence Testing and Analysis. One of the parameters of the method is the number of iterations, which is chosen by the operator. This number must be determined according to the accuracy required. The objective is to find a compromise between precision and computation time by evaluating the convergence of the algorithm as a function of the number of iterations. To do so, we have run the algorithm 10 times on different instances with a parameter value set to 1,000,000 iterations. Figure 6 shows the evolution of the mean value (over 10 runs for one instance) of the lost demand in function of the number of iterations (here represented up to 120,000 iterations).

It can be noticed that the curve stabilizes relatively quickly. On this instance, with an initial loss of 698 users, no
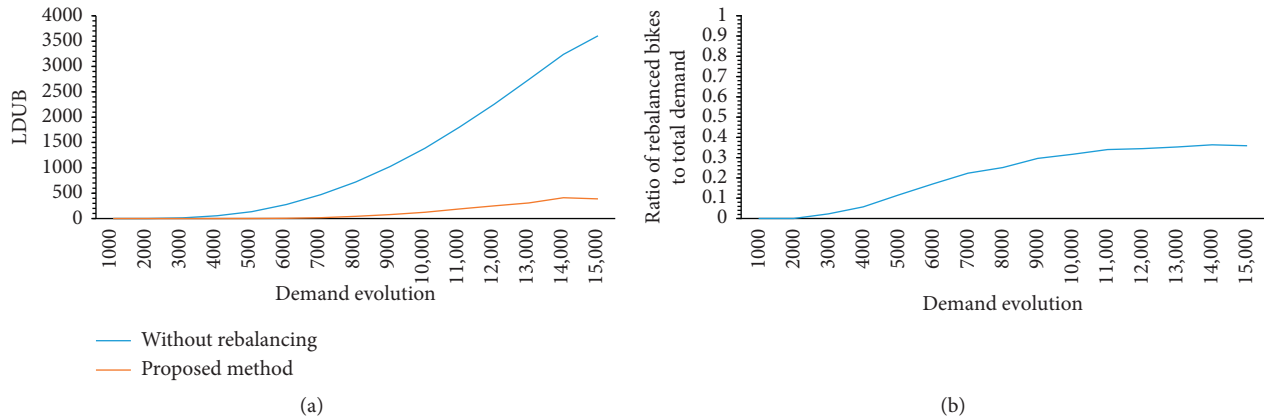
(a)

(b)

FIGURE 5: (a) LDUB evaluation as a function of demand evolution; (b) ratio of rebalanced bicycles to total demand.
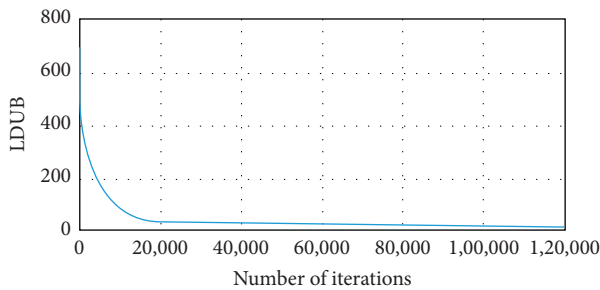

FIGURE 6: Evolution of average LDUB in function of iterations.

FIGURE 7: Evolution of LDUB in function of acceptance rate variation.

improvement was possible after a number of iterations close to 110,000. It can be noted then that the relative improvement over the last 20,000 iterations is only 1.3% on average per thousand iterations, which is relatively small. Depending on the instance, this stabilization point may occur earlier (around the 30,000[th] iteration on instance with a lower initial loss of 222 users). Therefore, subsequent iterations are potentially useless.

We have therefore set the number of iterations at 100,000. This corresponds to an execution time of about 3.5 minutes (on a PC running Windows 10 with an Intel Core™ i7-6700 CPU@6.40 GHz and 8 GB RAM).

5.5.2. Variation in Acceptance Rates. Acceptance rates assigned to neighbouring stations of an empty station influence the LDBU result, calculated by the proposed method. Like explained in the Acceptance Rate section, it was defined from a survey of 136 users of the Vélib' system. As this sample is not large enough, an analysis of the variation in acceptance rates was carried out in terms of the impact on the LDBU. This analysis shows the influence of these rates on the results of the proposed method.

A factor $\alpha \in \{0.3, 0.35, 0.4, \ldots, 0.95, 1, 1.05\}$ has been applied to all acceptance matrices of the system. The lower the value of α, the greater the number of users who will not be willing to go to a neighbouring station in case of bike shortage. Note that $\alpha \leq 1.05$, otherwise the highest acceptance rate would exceed 1 (the acceptance rate for Zone 1 is
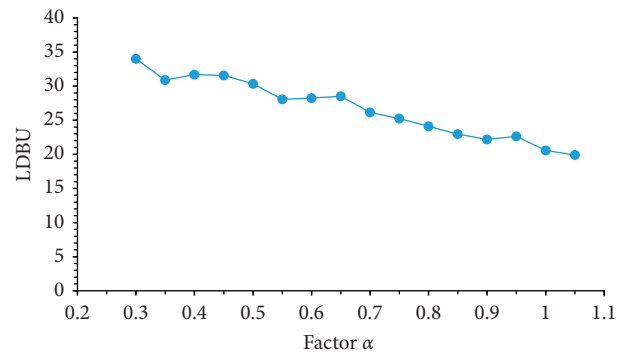
0.97). We have chosen to use the data of the third studied day (September 06th) because the LDUB of this day, calculated during Experiment 2, is very close to the average LDUB of the 29 studied days.

Figure 7 shows, as expected, that the LDUB increases when α decreases. Indeed, the acceptance rate decreases when α decreases, and thus fewer users move to a neighbouring station when the desired station is empty. If we compare LDUB for the values of α equal to 0.3 and 1, LDUB is 65% higher in the first case. Thus, the proposed method is more relevant for systems with higher acceptance rates.

A priori, the acceptance rates will be higher for systems with a small distance between stations. As these acceptance rates are a parameter of the proposed method, they can be modified by the system manager to better characterize the behaviour of users of the system.

6. Conclusion

In this article, we proposed a modelling of the user behaviour of a BSS (made up of stations and dockings). This behaviour concerns users who do not find an available bicycle in the desired station and can move to the nearest nonempty station to pick up a bicycle. Our main objective is to use this model to realistically evaluate the lost demand as a measure of user dissatisfaction by proposing an algorithm.

Our second objective is to show the feasibility of using this calculation method to support system control decisions, in particular static control. A coupling between two algorithms is performed. The optimization algorithm of the static rebalancing minimizes the number of lost demands. The lost demand is calculated by the algorithm simulating the user behaviour. The method defines a static rebalancing plan determining the stations and quantities of bicycles to be dropped off and the stations and quantities of bicycles to be picked up.

Three different experiments of the method have been carried out, and very interesting results concerning its use have been obtained on a real system (Vélib'). In particular, we found the following:

(i) Experiment 1 (evaluation of lost demand): the LDUB indicator, calculated by Algorithm 1 of our method, shows a deviation of around 36% on average compared to the more traditional lost demand indicator found in the literature.

(ii) Experiment 2 (static rebalancing of the Vélib' system): the rebalancing plan defined by the proposed method has an average LDUB improvement of around 82% compared with the rebalancing plan applied by the manager (JCDecaux). Moreover, this is done by moving less than 34% of bicycles on average.

(iii) Experiment 3 (forward-looking framework): in a case of gradual growth in demand (variation from 1000 to 15,000 bicycles), the proposed method never achieves demand losses (LDUB) greater than 3% of the total system demand, and the number of bicycles rebalanced does not exceed 36% of the total bicycle demand.

These results show, from an operational point of view, the interest of using this method, in order to be able to better dimension the resources to be implemented in order to manage a BSS more efficiently. It is important to note that the budget allocated to rebalancing operations is not very large (c.f. Section 1), requiring fewer resources to be deployed for its execution in order to achieve better customer satisfaction.

From a more strategic point of view, the third experiment shows that decision-makers can also use the proposed method in more prospective studies, such as the deployment of a new system and the restructuring or extension of an existing system. In this context, the tool would be applied as a decision support tool to test different scenarios and compare them on the basis of a more realistic indicator which is the LDUB.

Although the proposed method offers very good results, it has limitations. As it stands, the method has been applied to cases that considered the real system demand. However, operators make decisions using demand forecasts. Thus, for the results presented in this section to be reproducible, it is necessary for the decision-maker to be able to integrate reliable demand forecasts.

As regards the continuity of this study, the next step will concern evolutions related to user behaviour and to rebalancing plan. In respect of user behaviour, we will integrate into the proposed method other criteria for the choice of the neighbouring station made by the users (the station with the most bicycles, the station closer in the direction of the user's movement, etc.). In this way, user behaviour will be more realistic. Concerning the rebalancing plan, we will also take into account the docking demand and rebalancing to define it, as docking availability is necessary at the end of users' trips. In this case, the demand is not really lost; as for this type of BSS, the user is obliged to return the bicycle to a station. It is therefore necessary, for Algorithm 1, to take into account the transfer of the demand for the bicycle from a full station to a neighbouring station that is not full. This will allow us to create a rebalancing plan (by Algorithm 2) without having predefined a list of supplier stations.

Then, we will consider the problem of setting up rebalancing vehicle routes, taking into account the capacity constraints of the vehicles in order to minimize the total distance travelled. It would be possible to apply a vehicle routing optimization algorithm.

Secondly, it would also be interesting to develop the study towards dynamic rebalancing. This would imply the integration of temporal dimensions to the problem, such as the evolution of the demand for bicycles and dockings over time, the walking time of users to reach a neighbouring station, and the duration of rebalancing vehicle tours.

Concerning the solving method, other metaheuristics could be implemented, for example, based on simulated annealing, to avoid being trapped in a local minimum. In addition, other neighbourhood operator could be used in order to implement mechanisms for diversifying the neighbouring solutions obtained.

Data Availability

You can find the data used in this study and the source code of the proposed method in the link below: https://www.dropbox.com/sh/p220jhtpp5g72t3/AACXN-G7LC4bmRPjzNx2m9yqa?dl=0.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

References

[1] L. Zhang, J. Zhang, Z.-y. Duan, and D. Bryde, "Sustainable bike-sharing systems: characteristics and commonalities across cases in urban China," *Journal of Cleaner Production*, vol. 97, pp. 124–133, 2015.

[2] I. Ţuţu, C. E. Chiriţă, A. Lopes, and J. L. Fiadeiro, "Logical support for bike-sharing system design," *Software Engineering*

to Formal Methods and Tools, and Back, Springer, Berlin, Germany, 2019.

[3] E. Fishman, "Bikeshare: a review of recent literature," *Transport Reviews*, vol. 36, no. 1, pp. 92–113, 2016.

[4] H. Si, J.-g. Shi, G. Wu, J. Chen, and X. Zhao, "Mapping the bike sharing research published from 2010 to 2018: a scientometric review," *Journal of Cleaner Production*, vol. 213, pp. 415–427, 2019.

[5] B. Beroud and E. Anaya, "Chapter 11 private interventions in a public service: an analysis of public bicycle schemes," *Cycling and Sustainability*, vol. 1, pp. 269–301, 2012.

[6] E. Fishman, S. Washington, and N. Haworth, "Bike share's impact on car use: evidence from the United States, Great Britain, and Australia," *Transportation Research Part D: Transport and Environment*, vol. 31, pp. 13–20, 2014.

[7] P. Jensen, J.-B. Rouquier, N. Ovtracht, and C. Robardet, "Characterizing the speed and paths of shared bicycle use in Lyon," *Transportation Research Part D: Transport and Environment*, vol. 15, no. 8, pp. 522–524, 2010.

[8] Z. Wang, L. Cheng, Y. Li, and Z. Li, "Spatiotemporal characteristics of bike-sharing usage around rail Transit stations: evidence from Beijing, China," *Sustainability*, vol. 12, no. 4, p. 1299, 2020.

[9] R. Zhao, L. Yang, X. Liang, and G. Anfu, "Last-mile travel mode choice: data-mining hybrid with multiple attribute decision making," *Sustainability*, vol. 11, no. 23, pp. 1–15, 2019.

[10] P. Midgley, *Bicycle-Sharing Schemes: Enhancing Sustainable Mobility in Urban Areas*, Vol. 8, United Nations Department of Economic and Social Affairs, New York, NY, USA, 2011.

[11] Y. Ji, X. Ma, M. He, Y. Jin, and Y. Yuan, "Comparison of usage regularity and its determinants between docked and dockless bike-sharing systems: a case study in Nanjing, China," *Journal of Cleaner Production*, vol. 255, Article ID 120110, 2020.

[12] M. Loidl, U. Witzmann-Müller, and B. Zagel, "A spatial framework for planning station-based bike sharing systems," *European Transport Research Review*, vol. 11, no. 1, p. 9, 2019.

[13] S. Martínez, A. Tapia, V. Bernardo, J. E. Ricart, and M. R. Planas, *The Economic Impact of Bike Sharing in European Cities*, IESE, Barcelona, Spain, 2019.

[14] Z. Chen, D. van Lierop, and D. Ettema, "Dockless bike-sharing systems: what are the implications?" *Transport Reviews*, vol. 40, no. 3, pp. 333–353, 2020.

[15] S. Ghosh, J. Y. Koh, and P. Jaillet, "Improving customer satisfaction in bike sharing systems through dynamic repositioning," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pp. 5864–5870, Macao, China, August 2019.

[16] C. M. De Chardon, G. Caruso, and I. Thomas, "Bike-share rebalancing strategies, patterns, and purpose," *Journal of Transport Geography*, vol. 55, pp. 22–39, 2016.

[17] Y. Liu, W. Y. Szeto, and S. C. Ho, "A static free-floating bike repositioning problem with multiple heterogeneous vehicles, multiple depots, and multiple visits," *Transportation Research Part C: Emerging Technologies*, vol. 92, pp. 208–242, 2018.

[18] F. Chiariotti, C. Pielli, A. Zanella, and M. Zorzi, "A dynamic approach to rebalancing bike-sharing systems," *Sensors*, vol. 18, no. 2, p. 512, 2018.

[19] A. Singla, M. Santoni, G. Bartók, P. Mukerji, M. Meenen, and A. Krause, "Incentivizing users for balancing bike sharing systems," in *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, Austin, TX, USA, January 2015.

[20] J. H. Lin and T. C. Chou, "A geo-aware and VRP-based public bicycle redistribution system," *International Journal of Vehicular Technology*, vol. 2012, Article ID 963427, 2012.

[21] Institute for Transportation & Development Policy, *The Bike-Share Planning Guide*, ITDP, New York, NY, USA, 2013.

[22] A. Faghih-Imani and N. Eluru, "Incorporating the impact of spatio-temporal interactions on bicycle sharing system demand: a case study of New York CitiBike system," *Journal of Transport Geography*, vol. 54, pp. 218–227, 2016.

[23] D. Gavalas, C. Konstantopoulos, and G. Pantziou, "Design and management of vehicle-sharing systems: a survey of algorithmic approaches," in *Smart Cities and Homes*, pp. 261–289, Elsevier, Amsterdam, Netherlands, 2016.

[24] P. Vogel and D. C. Mattfeld, "Strategic and operational planning of bike-sharing systems by data mining-a case study," *Lecture Notes in Computer Science*, pp. 127–141, 2011.

[25] E. Fishman and P. Schepers, "Global Bikeshare: what the data tells us about safety," in *Proceedings of the 3rd International Cycling Safety Conference (ICSC2014)*, Gothenburg, Sweden, 2014.

[26] S. C. Ho and W. Y. Szeto, "Solving a static repositioning problem in bike-sharing systems using iterated tabu search," *Transportation Research Part E: Logistics and Transportation Review*, vol. 69, pp. 180–198, 2014.

[27] T. Raviv, M. Tzur, and I. A. Forma, "Static repositioning in a bike-sharing system: models and solution approaches," *EURO Journal on Transportation and Logistics*, vol. 2, no. 3, pp. 187–229, 2013.

[28] H. M. Espegren, J. Kristianslund, H. Andersson, and K. Fagerholt, "The static bicycle repositioning problem-literature survey and new formulation," in *Proceedings of the International Conference on Computational Logistics*, pp. 337–351, Padova, Italy, June 2016.

[29] D. Chemla, F. Meunier, and R. Wolfler Calvo, "Bike sharing systems: solving the static rebalancing problem," *Discrete Optimization*, vol. 10, no. 2, pp. 120–146, 2013.

[30] G. Erdoğan, M. Battarra, and R. Wolfler Calvo, "An exact algorithm for the static rebalancing problem arising in bicycle sharing systems," *European Journal of Operational Research*, vol. 245, no. 3, pp. 667–679, 2015.

[31] F. Cruz, A. Subramanian, B. P. Bruck, and M. Iori, "A heuristic algorithm for a single vehicle static bike sharing rebalancing problem," *Computers & Operations Research*, vol. 79, pp. 19–33, 2017.

[32] P. Papazek, C. Kloimüllner, B. Hu, and G. R. Raidl, "Balancing bicycle sharing systems: an analysis of path relinking and recombination within a GRASP hybrid," in *Parallel Problem Solving from Nature-PPSN XIII*, Springer, Berlin, Germany, 2014.

[33] M. Rainer-Harbach, P. Papazek, B. Hu, and G. R. Raidl, "Balancing bicycle sharing systems: a variable neighborhood search approach," in *Evolutionary Computation in Combinatorial Optimization*, Springer, Berlin, Germany, 2013.

[34] C. Rudloff and B. Lackner, "Modeling demand for bikesharing systems: neighboring stations as source for demand and reason for structural Breaks," *Transportation Research Record*, vol. 2430, no. 1, pp. 1–11, 2014.

[35] M. Rainer-Harbach, P. Papazek, G. R. Raidl, B. Hu, and C. Kloimüllner, "PILOT, GRASP, and VNS approaches for the static balancing of bicycle sharing systems," *Journal of Global Optimization*, vol. 63, no. 3, pp. 597–629, 2015.

[36] S. Zhang, G. Xiang, and Z. Huang, "Bike-sharing static rebalancing by considering the collection of bicycles in need

of repair," *Journal of Advanced Transportation*, vol. 2018, Article ID 8086378, 2018.

[37] L. D. Gaspero, A. Rendl, and T. Urli, "Balancing bike sharing systems with constraint programming," *Constraints*, vol. 21, no. 2, pp. 318–348, 2016.

[38] I. A. Forma, T. Raviv, and M. Tzur, "A 3-step math heuristic for the static repositioning problem in bike-sharing systems," *Transportation Research Part B: Methodological*, vol. 71, pp. 230–247, 2015.

[39] C. Lv, C. Zhang, K. Lian, Y. Ren, and L. Meng, "A hybrid algorithm for the static bike-sharing re-positioning problem based on an effective clustering strategy," *Transportation Research Part B: Methodological*, vol. 140, pp. 1–21, 2020.

[40] M. Dell'Amico, M. Iori, S. Novellani, and T. Stützle, "A destroy and repair algorithm for the Bike sharing Rebalancing Problem," *Computers & Operations Research*, vol. 71, pp. 149–162, 2016.

[41] M. Kaspi, T. Raviv, and M. Tzur, "Parking reservation policies in one-way vehicle sharing systems," *Transportation Research Part B: Methodological*, vol. 62, pp. 35–50, 2014.

[42] A. A. Kadri, I. Kacem, and K. Labadi, "A branch-and-bound algorithm for solving the static rebalancing problem in bicycle-sharing systems," *Computers & Industrial Engineering*, vol. 95, pp. 41–52, 2016.

[43] F. Chiariotti, C. Pielli, A. Zanella, and M. Zorzi, "A bike-sharing optimization framework combining dynamic rebalancing and user incentives," *ACM Transactions on Autonomous and Adaptive Systems*, vol. 14, no. 3, 2020.

[44] J. April, F. Glover, J. P. Kelly, and M. Laguna, "Practical introduction to simulation optimization," in *Proceedings of the 35th Conference on Winter Simulation: Driving Innovation*, pp. 71–78, New Orleans, LA, USA, December 2003.

[45] R. Costa Affonso and F. Couffin, "Aggregated indicator for assessing station criticality of bike sharing systems," in *Proceedings of the 2019 International Conference on Industrial Engineering and Systems Management (IESM)*, pp. 1–6, Shanghai, China, September 2019.

[46] Y. Ai, Z. Li, and M. Gan, "A solution to measure traveler's transfer tolerance for walking mode and dockless bike-sharing mode," *The Journal of Supercomputing*, vol. 75, no. 6, pp. 3140–3157, 2019.

[47] A. Negahban, "Simulation-based estimation of the real demand in bike-sharing systems in the presence of censoring," *European Journal of Operational Research*, vol. 277, no. 1, pp. 317–332, 2019.

[48] S. Albiński, P. Fontaine, and S. Minner, "Performance analysis of a hybrid bike sharing system: a service-level-based approach under censored demand observations," *Transportation Research Part E: Logistics and Transportation Review*, vol. 116, pp. 59–69, 2018.

[49] E.-G. Talbi, *Metaheuristics: From Design to Implementation*, John Wiley & Sons, Hoboken, NJ, USA, 2009.