

Research Article

Rebalancing Strategy for Bike-Sharing Systems Based on the Model of Level of Detail

Zhenghua Hu ^{1,2}, Kejie Huang ¹, Enyou Zhang ³, Qi'ang Ge ², and Xiaoxue Yang ²

¹College of Information Science & Electronic Engineering, Zhejiang University, Hangzhou, China

²School of Electronic and Information Engineering, Ningbo University of Technology, Ningbo, China

³Ningbo Jianan Electronics Co., Ltd, Cixi, China

Correspondence should be addressed to Kejie Huang; huangkejie@zju.edu.cn

Received 2 July 2021; Revised 10 August 2021; Accepted 14 September 2021; Published 27 September 2021

Academic Editor: Chi-Hua Chen

Copyright © 2021 Zhenghua Hu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Traveling by bike-sharing systems has become an indispensable means of transportation in our daily lives because green commuting has gradually become a consensus and conscious action. However, the problem of “difficult to rent or to return a bike” has gradually become an issue in operating the bike-sharing system. Moreover, scientific and systematic schemes that can efficiently complete the task of rebalancing bike-sharing systems are lacking. This study aims to introduce the basic idea of the k -divisive hierarchical clustering algorithm. A rebalancing strategy based on the model of level of detail in combination with genetic algorithm was proposed. Data were collected from the bike-sharing system in Ningbo. Results showed that the proposed algorithm could alleviate the problem of the uneven distribution of the demand for renting or returning bikes and effectively improve the service from the bike-sharing system. Compared with the traditional method, this algorithm helps reduce the effective time for rebalancing bike-sharing systems by 28.3%. Therefore, it is an effective rebalancing scheme.

1. Introduction

The issue of traffic congestion is becoming increasingly serious with the increase in the number of motor vehicles in urban areas [1]. This situation has become a major problem in the development of modern cities. In many big cities in China, the running speed of cars on the arterial roads during rush hours is less than 20 km/h. Persistent traffic congestion and a series of social problems, such as environmental pollution, energy waste, and exhaust emissions, have seriously hindered the development of economy. Nevertheless, people are paying increasing attention to environmental protection, energy saving, and emission reduction because of the increasing transportation problems and the deteriorating ecological environment. The concept of green commuting has also attracted increasing attention, and the vigorous development of a public transport system in cities has become the community consensus. Traveling by bike-sharing system is regarded as a green and low-carbon way of transportation [2]. This mode of transport can alleviate the

traffic pressure on the road and solve “last mile” problem in public transportation [3]. Therefore, traveling by bike-sharing system has become increasingly popular and received the attention of governments worldwide [4–7].

However, bikes cannot spontaneously form a relatively balanced distribution due to the asymmetry of traffic flow and the uncertainty of needs from users. The government has utilized a considerable proportion of manpower and financial resources to rebalance bikes. However, situations such as “no bike to rent” or “no pile to return a bike” still exist in many stations due to the hysteresis and non-scientificity of the rebalance [8, 9]. Consequently, many citizens abandon bike-sharing system when they go out, thereby greatly reducing the probabilities for people to use bike-sharing system [10–12] and restricting the development of bike-sharing systems to a certain extent.

Simply increasing facilities to improve the service of the bike-sharing system and increase the satisfaction from the public cannot fundamentally solve the problem. How to effectively rebalance bikes among stations to ensure that

stations with fewer bikes can be quickly transferred in, whereas those with more bikes can be transferred out, has become a topic of common concern for more and more researchers [13, 14].

This study develops a rebalancing scheme based on the model of level of detail and the hierarchy of cognitive processes. This method performs the rebalancing scheme with granularity from coarse to fine throughout the whole experimental area. By taking the bike-sharing system in Ningbo city as an example, this study verified through a comparative experiment that the rebalancing scheme proposed can improve the operational efficiency of the bike-sharing system and reduce the management cost. Such an undertaking is performed to support the related departments in formulating reasonable rebalancing schemes for bike-sharing system.

2. Literature Review

2.1. Rebalancing Schemes. In recent years, researches related to bike-sharing system have been advancing, and these studies have provided a large number of ideas for rebalancing bike-sharing system. The main options to rebalance bike-sharing system include user-based and operator-based strategy [15]. The former is more popular in dockless bike-sharing system, while the latter performs better in station-based bike-sharing system. For the latter, current trend shows a hybrid solution in combination with different rebalancing modes.

Yin et al. [16] developed a logit model to estimate the influence of sociodemographic, land-use, and public transit characteristics on rebalancing bike-sharing system and got good result. But the logit model is too simple to deal with the complex pattern in rebalancing bike-sharing system. Lu [17] treated the issue of providing service to as much users as possible as a bike-dock pair allocation problem. They further developed an algorithm with local ratio to solve the problem. Results showed that the algorithm could save time for trips and help more users to use bike-sharing system. But if users are unwilling to walk a long way to the station, the algorithm would not work. Shi [18] designed an improved particle swarm optimization algorithm and formulated a corresponding model for rebalancing bike-sharing. However, the trucks generally have a low ratio of the average load according to the statistics. So, the algorithm needs further optimization. Liu and Ren [19] proposed a dynamic rebalancing strategy with an improved genetic algorithm. This algorithm aims to improve the overall efficiency by reducing the number of routes. But some idealized settings have caused a lot of errors in the model and reduced the effect. Mao [20] proposed a dynamic rebalancing model for bike-sharing system and calculated the number of bikes in each area with spatiotemporal graph to analyze the mobility patterns. Results showed that the proposed method had high accuracy. But there are still factors unconsidered in the model which will affect the accuracy. Recently, some scholars [21] proposed a bottom-up cluster-based model to rebalance bike-sharing system in combination with the

spatiotemporal characteristic of the bike-sharing trips. However, this method is purely a static rebalancing strategy, and its effect and performance need to be improved.

2.2. Model of Level of Detail. The model of level of detail (LoD) originated in the field of computer vision. The computer determines the allocation of resources to render objects according to the position and importance in the environment so as to reduce the level of detail of non-important objects and obtains higher rendering efficiency. In recent years, the model of level of detail has been widely used in many fields.

Fan et al. [22] proposed a hierarchy training strategy on images in classification. The network would be more stable and robust through hierarchy training. Experiments show that the network trained by this strategy will get better result. Wang [23] presented a reliability-oriented hierarchical control strategy. This strategy is adopted to further improve the efficiency of the microgrids. Results show that the proposed hierarchical control structure has better performance towards reliability enhancement. The image pyramid model is another application of the level of detail model in the storage and management of the images [24]. The image pyramid establishes a series of layers with different resolutions and establishes a corresponding spatial indexing mechanism. This will improve the rendering speed of zooming or browsing images. Hu [25] further introduced a routing strategy with hierarchical model and calculated the optimal travel route with a hierarchical manner. Nakano [26] classified the transposable elements as a hierarchical classification issue and proposed hierarchical classification strategies with new data sets. Experimental results showed that the method proposed had a more competitive result compared to those by other strategies.

In short, the data structure of the model of level of detail is easy and clear. It has been applied in many academic fields. Meanwhile, the implementation of the model is simple, so it is more suitable for rebalancing bike-sharing system in urban areas.

2.3. Modeling the Rebalance of Bike-Sharing Systems

2.3.1. Overview. To study rebalancing bike-sharing system is to determine the stations that need to be rebalanced and the number of bikes that must be transferred in or out of each station, as well as to compute the optimal path for rebalancing trucks [27].

This study assumes that the total numbers of bikes and the stakes in each station are limited to simplify the rebalancing problem. A bike is either locked at a station or being used by a user. Three states occur in each station, namely, no bikes available, no idle piles, and available bikes and idle piles. We assumed that a certain number of rebalancing centers are set up. These centers are in charge of rebalancing bikes for the stations around them. Each center is equipped with several trucks. The carrying capacity of these trucks remains constant. The rebalancing system will always

monitor the status of the bike-sharing system. When the locked ratio of the piles at the station is lower than or exceeds a certain range, the system would automatically alert the staff at the management center to rebalance the bikes.

2.3.2. Model Construction. Therefore, how to design a reasonable and efficient rebalancing scheme based on the demand from stations has been a key issue. With this scheme, trucks can pass through stations in an orderly manner and finally return to the center. In this situation, the bike-sharing system could meet the needs from users as much as possible and improve the service of bike-sharing system [28, 29]. Moreover, the cost of rebalancing bike-sharing system must be taken into account, so that the running distance by the trucks is the shortest or the total time spent is the least. This is a problem that urgently needs to be solved in the current bike-sharing system, which is also the problem that this study tends to solve.

Therefore, the rebalancing scheme proposed in this work takes the transportation cost and satisfaction from bike-sharing system as the objective functions.

(1) Transportation Cost. The truck travel must be short to minimize transportation costs. The distance covered by a truck for performing rebalance for a trip is shown in the following formula:

$$Z_1 = \sum_{i=0}^n \sum_{\substack{j=0 \\ i \neq j}}^n X_{ij} d_{ij} \omega, \quad (1)$$

where X_{ij} is a decision variable. $X_{ij} = 1$ means that the truck moves from station i to station j in a certain trip, while $X_{ij} = 0$ means that the truck does not travel from station i to station j . d_{ij} is the distance from station i to station j , and ω

is the transportation cost per kilometer. N refers to the total number of the stations that need to be rebalanced in a certain trip.

(2) Satisfaction. Satisfaction can be translated into penalty costs that do not satisfy time windows. Specifically, the dispatching truck should complete the dispatch task within the time expected as soon as possible to ensure that users can avail of the services at the station. We use soft-time window to constrain the arrival time of the trucks. The penalty cost is high when the arrival time before/after the expected time is early/late. The penalty cost of the time is shown in the following formula:

$$Z_2 = \sum_{i=1}^n W_i (epu \cdot \max(L_i - t_i, 0) + lpu \cdot \max(t_i - U_i, 0)), \quad (2)$$

where W_i is a decision variable indicating whether station i needs to be rebalanced. t_i is the time when the truck arrives at station i ; $[L_i, U_i]$ is the time window of site i , and L_i represents the earliest time expected by site i ; U_i is the latest time expected by site i ; epu is the penalty cost for early arrival when the time window is unsatisfied; and lpu is the penalty cost for late arrival. Figure 1 shows the diagram of the penalty function for the soft-time window.

In summary, the objective function for the rebalancing model is shown as follows:

$$T = \alpha Z_1 + \beta Z_2, \quad (3)$$

where α is the weight value of the transportation cost, β is the weight value of penalty cost of time window, Z_1 is the transportation cost, and Z_2 is the penalty cost of the time window. Thus, the objective function to be optimized is shown as follows:

$$\alpha \left(\sum_{i=0}^n \sum_{\substack{j=0 \\ i \neq j}}^n X_{ij} d_{ij} \omega \right) + \beta \left(\sum_{i=1}^n W_i \times (epu \cdot \max(L_i - t_i, 0) + lpu \cdot \max(t_i - U_i, 0)) \right) \longrightarrow \text{Min}. \quad (4)$$

The constraints of the above formula are as follows:

(1) Decision variable is a 0-1 variable, which is defined as follows:

$$X_{ij} = \begin{cases} 1 & \text{if the rebalancing truck runs from station } i \text{ to } j \\ 0 & \text{otherwise} \end{cases}. \quad (5)$$

(2) In each trip, every station can only be served once by a truck. Thus,

$$\sum_{\substack{i=0 \\ i \neq j}}^n X_{ij} = 1, \quad \forall j \in V \setminus \{0\}, \quad (6)$$

$$\sum_{\substack{j=0 \\ i \neq j}}^n X_{ij} = 1, \quad \forall i \in V \setminus \{0\}, \quad (7)$$

where V represents the collection of all the stations, including the rebalancing center which is

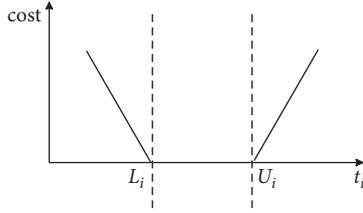


FIGURE 1: Penalty cost for the soft-time window.

represented by $\{0\}$. The rebalancing center can be visited multiple times in each trip.

- (3) The truck would visit bike stations one by one.

$$\sum_{\substack{i=0 \\ i \neq p}}^n X_{ip} = \sum_{\substack{j=0 \\ j \neq p}}^n X_{pj}, \quad \forall p \in V \setminus \{0\}. \quad (8)$$

- (4) The loading of each truck must be a nonnegative integer and does not exceed its maximum capacity.

$$\begin{aligned} g_{ij} &\in \mathbb{Z}^+, \\ g_{ij} &\leq Q \times X_{ij} \quad \forall i, j \in V, i \neq j, \end{aligned} \quad (9)$$

where g_{ij} represents the number of bikes that the truck transports from station i to station j , and Q is the maximum number of bikes that the truck can carry.

- (5) The number of bikes is balanced before and after the station is rebalanced.

$$\sum_{\substack{j=0 \\ j \neq i}}^n g_{ji} + p_i = \sum_{\substack{j=0 \\ j \neq i}}^n g_{ij} + q_i, \quad \forall i \in V \setminus \{0\}, \quad (10)$$

where p_i is the initial number of the bikes before the station is rebalanced, and q_i is the number afterward.

- (6) The arrival time t_j of the truck at station j is shown in the following formula:

$$t_j = \sum_{i=0}^n X_{ij} (t_i + tt_{ij} + |p_i - q_i| \times wt), \quad j \in V \setminus \{0\}, \quad (11)$$

where wt is the time of loading or unloading a bike, and tt_{ij} is the running time of the truck from station i to station j .

- (7) The truck shall visit the station one by one, as shown in formula (12), where M is a number that approaches infinity.

$$t_i + tt_{ij} + (1 - X_{ij}) \times M \leq t_j, \quad \forall i \in V, j \in V. \quad (12)$$

In summary, the model of the rebalancing algorithm for bike-sharing system proposed in this study is shown as follows:

$$\begin{aligned} &\alpha \left(\sum_{i=0}^n \sum_{\substack{j=0 \\ j \neq i}}^n X_{ij} d_{ij} \omega \right) + \beta \left(\sum_{i=1}^n W_i \times (epu \cdot \max(L_i - t_i, 0) + lpu \cdot \max(t_i - U_i, 0)) \right) \longrightarrow \text{Min}, \\ \text{st.} &\left\{ \begin{array}{l} \sum_{i=0}^n X_{ij} = 1 \quad \forall j \in V \setminus \{0\} \\ \sum_{j=0}^n X_{ij} = 1 \quad \forall i \in V \setminus \{0\} \\ \sum_{\substack{i=0 \\ i \neq p}}^n X_{ip} = \sum_{\substack{j=0 \\ j \neq p}}^n X_{pj} \quad \forall p \in V \setminus \{0\} \\ g_{ij} \in \mathbb{Z}^+ \text{ and } g_{ij} \leq Q \times X_{ij} \quad \forall i, j \in V, i \neq j \\ \sum_{\substack{j=0 \\ j \neq i}}^n g_{ji} + p_i = \sum_{\substack{j=0 \\ j \neq i}}^n g_{ij} + q_i \quad \forall i \in V \setminus \{0\} \\ t_j = \sum_{i=0}^n X_{ij} (t_i + tt_{ij} + |p_i - q_i| \times wt) \quad j \in V \setminus \{0\} \\ t_i + tt_{ij} + (1 - X_{ij}) \times M \leq t_j \quad \forall i \in V, j \in V \\ X_{ij} = \{0, 1\} \end{array} \right. \end{aligned} \quad (13)$$

2.4. K-Means. Clustering is an unsupervised learning method in machine learning. This method uses the similarity between objects to divide the entire data set into several different clusters. Such step is conducted to ensure that the data in the same cluster has a great degree of similarity, and the similarity among data in different clusters is low. In this manner, clustering is often used to reveal the inherent nature among data. The topology relationship of stations is an important spatial feature because the distribution of bike stations is regarded as a network topological structure. A reasonable division scheme is easy to design when dividing the areas where stations locate if the characteristic of the network topology is considered. In this study, we use the K-means clustering algorithm to cluster the stations on the basis of the spatial characteristics of the stations.

K-means algorithm is a commonly used clustering algorithm and an important analytical tool in research fields. This algorithm aims to randomly select k points in the data set as the initial cluster centers and then calculate the distance between each center and the remaining objects. The remaining data are then divided into the clusters closest to these k centers. Finally, the center of each cluster is recalculated according to the objects in the cluster, and the cluster is redivided. This process is repeated until the result of each division remains the same or the sum of the squared errors of the points in each cluster reaches the minimum, and the algorithm ends.

In K-means, the selection of the k value directly affects the final clustering results. Many strategies in the literature can be adopted to determine the k value. In this work, we will use the elbow rule. The core index of the elbow rule is the sum of squared errors (SSE). This index indicates the quality of the clustering result. The degree of aggregation of the clusters would increase and the value of SSE would naturally become smaller as the value of k increases [30]. Specifically, if the value of k is smaller than the real number of the clusters, the decline in SSE would be large since the increase of k will increase the aggregation of each cluster. However, if k becomes closer to the real number of the clusters, the reward in increasement of the aggregation will decrease sharply [30]. Accordingly, the decline of SSE will be sharply reduced. As the value of k continues to increase, it tends to become flattened. This indicates that the relationship between SSE and k looks like the shape of an elbow. The value of k that corresponds to the elbow part is the best clustering number, as shown in Figure 2.

The curve in Figure 2 is similar to an elbow, and the k value corresponding to the elbow is the appropriate value.

2.5. Genetic Algorithm. Genetic algorithm (GA) is used to “search for the optimal solution by simulating the natural evolution process” [31, 32]. Some of the core operations in the genetic algorithm will be briefly introduced in this part.

2.5.1. Coding. A sequence of numbers are used for encoding to overcome such shortcomings as large search space and long codes generated by binary encoding. In this article, the number 0 is used to represent the rebalancing center and

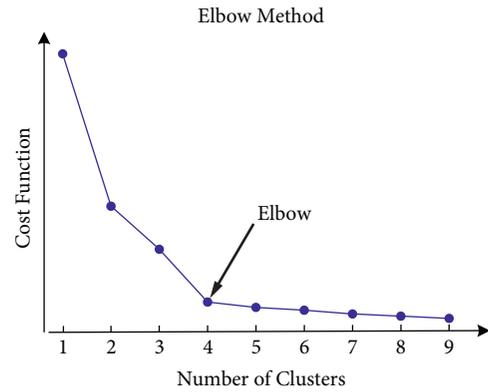


FIGURE 2: The elbow method.

$\{1, 2, 3, \dots, n\}$ to represent the bike stations that need to be rebalanced. For example, if a chromosome is $[0, 2, 3, 4, 0, 1, 5, 0]$, it means that the rebalancing truck departs from the center, rebalances stations 2, 3, and 4 in turn, and returns to the center. Then it departs from the center to rebalance stations 1 and 5 and returns to the center again. The specific route is as follows:

Rebalancing center \rightarrow station 2 \rightarrow station
 3 \rightarrow station 4 \rightarrow rebalancing center \rightarrow station
 1 \rightarrow station 5 \rightarrow rebalancing center

2.5.2. Initial Population. First, the numbers of all the stations that need to be rebalanced are randomly sorted, and then $k-1$ 0s are randomly inserted into the chromosome, where k is the number of rebalancing trucks. Then a 0 is inserted, respectively, at the beginning and at the end of the sequence, representing that the rebalancing trucks depart from the center and finally return to the center. After that, it is judged whether the current rebalancing scheme meets the capacity constraint of the truck and whether there are continuous zeros in the current rebalancing scheme. If the load constraint meets the requirement and there are no consecutive 0s in the rebalancing path, it will be chosen as an initial chromosome or it will be discarded and a new chromosome is regenerated. The above process is repeated to generate N chromosomes to form the initial population.

2.5.3. Fitness Function. Fitness can be used to simply evaluate the quality of a chromosome. The greater the value of the fitness function is, the better the corresponding chromosome is. Therefore, in this study, the reciprocal of the value of the cost function above is taken as the fitness function; namely,

$$\text{fit}(i) = \frac{1}{Z} \quad (14)$$

2.5.4. Selection. The elite retention and roulette wheel selection are combined in this paper, which not only ensure that the best individual survives into the next

generation but also enable individuals with large value of the fitness function to enter the next generation. It is assumed that the number of chromosomes in the current population is NP, and the specific operation steps are as follows:

- (1) The optimal individual with the greatest value of fitness function in the current population is recorded as *elite*. This individual does not perform pairwise crossover and mutation but is directly copied to the next generation.
- (2) The roulette wheel selection is performed on the remaining individuals. In other words, the NP-1 chromosomes are selected for crossover.

2.5.5. Crossover. The crossover operation is performed on the NP-1 chromosomes selected in the previous step. New individuals are continuously generated through crossover, expanding the search space of the algorithm .

- (1) When the number of the rebalancing trucks $K > 1$, this study adopts an improved crossover operator. Specific steps are shown in Figure 3.

Step 1: gene fragments with 0 at the beginning and at the end are randomly selected on the chromosomes of parent 1 and parent 2, respectively.

Step 2: prepend the gene other than the last 0 in the two gene fragments to the head of the parent chromosome.

Step 3: based on gene fragment 1 and gene fragment 2, offspring 1 and offspring 2 are constructed.

Step 4: expand offspring 1: traverse the genes of parent 2 from the left to the right, and add the genes that do not appear in gene segment 1 (offspring 1) to offspring 1 in order, and finally add code 0 at the end of offspring 1. Fill in offspring 2 symmetrically in the same way.

Step 5: for offspring 1, $K-2$ zeros are randomly inserted after segment 1 to ensure that there are K rebalancing paths. After that, it is judged whether the current chromosome meets the capacity constraint of the rebalancing truck and ensured that there are no consecutive 0s. If the constraints are not met, the $K-2$ zeros are reinserted. Offspring 2 is processed in the same way.

- (2) When the number of the rebalancing trucks $K = 1$, the cyclic crossover operator is used, and the specific steps are shown in Figure 4.

Step 1: first, the 0s at the beginning and at the end of parent 1 and parent 2 are deleted, and then a position is randomly selected on parent 1.

Step 2: find the number of the genes on the corresponding position in parent 2; and then return to parent 1 to find the gene position with the same number. Repeat this process until a ring is formed. The gene and the corresponding position of parent 1 in the ring are saved.

Step 3: the gene selected in parent 1 is picked to generate offspring 1, and the position order is not disturbed.

Step 4: the remaining genes in parent 2 are put into offspring 1.

Finally, add 0 to both ends of the chromosome to get the final result of chromosome of offspring 1. Similarly, chromosome of offspring 2 can be obtained.

2.5.6. Mutation. In this study, swapping mutation is selected to perform mutation on chromosomes; namely, two nonzero genes are randomly selected in the parent chromosomes and exchanged. If the route could meet the constraint of truck capacity after the exchange, it will be updated.

In summary, the complete algorithm flow of genetic algorithm is shown in Figure 5.

3. Rebalancing Strategy Based on the Model of Level of Detail

3.1. K-Divisive Hierarchical Clustering Algorithm. The divisive hierarchical clustering algorithm [33, 34] is also called the top-down clustering method. In this algorithm, all objects in the collection are treated as a cluster first. This cluster is then divided into small ones, each as a new cluster to be subdivided. Overall, the implementation of the divisive hierarchical clustering method is also a process of constructing a tree from top to bottom, as shown in Figure 6, where lines with different colors refer to different clusters. In the aforementioned recursive division process, all objects are first treated as the root node of a tree. The cluster is then recursively split into multiple small ones. This process is iteratively executed until only one object is present in each cluster or a predefined termination condition is satisfied.

Among the many divisive hierarchical clustering algorithms, the divisive hierarchical clustering with K-means, which is called the K-divisive hierarchical clustering, is one of the most widely used algorithms. This algorithm aims to divide data by using the divisive hierarchical clustering algorithms through iteratively applying K-means until each object becomes a cluster or the termination conditions are satisfied.

The K-divisive hierarchical clustering method combines the traditional K-means algorithm with divisive hierarchical clustering, thereby not only making the clustering realistic but also greatly improving the clustering efficiency [35].

3.2. Rebalancing Algorithm Based on the K-Divisive Hierarchical Clustering. The corresponding rebalancing strategy for bike-sharing system is designed in this section on the basis of the K-divisive hierarchical clustering algorithm. The previous analysis indicates that the K-divisive hierarchical clustering algorithm is actually a process of dividing the cluster with granularity from coarse to fine. The essence of the rebalancing algorithm proposed is a rebalancing strategy with granularity from large to small and a continuous refinement process.

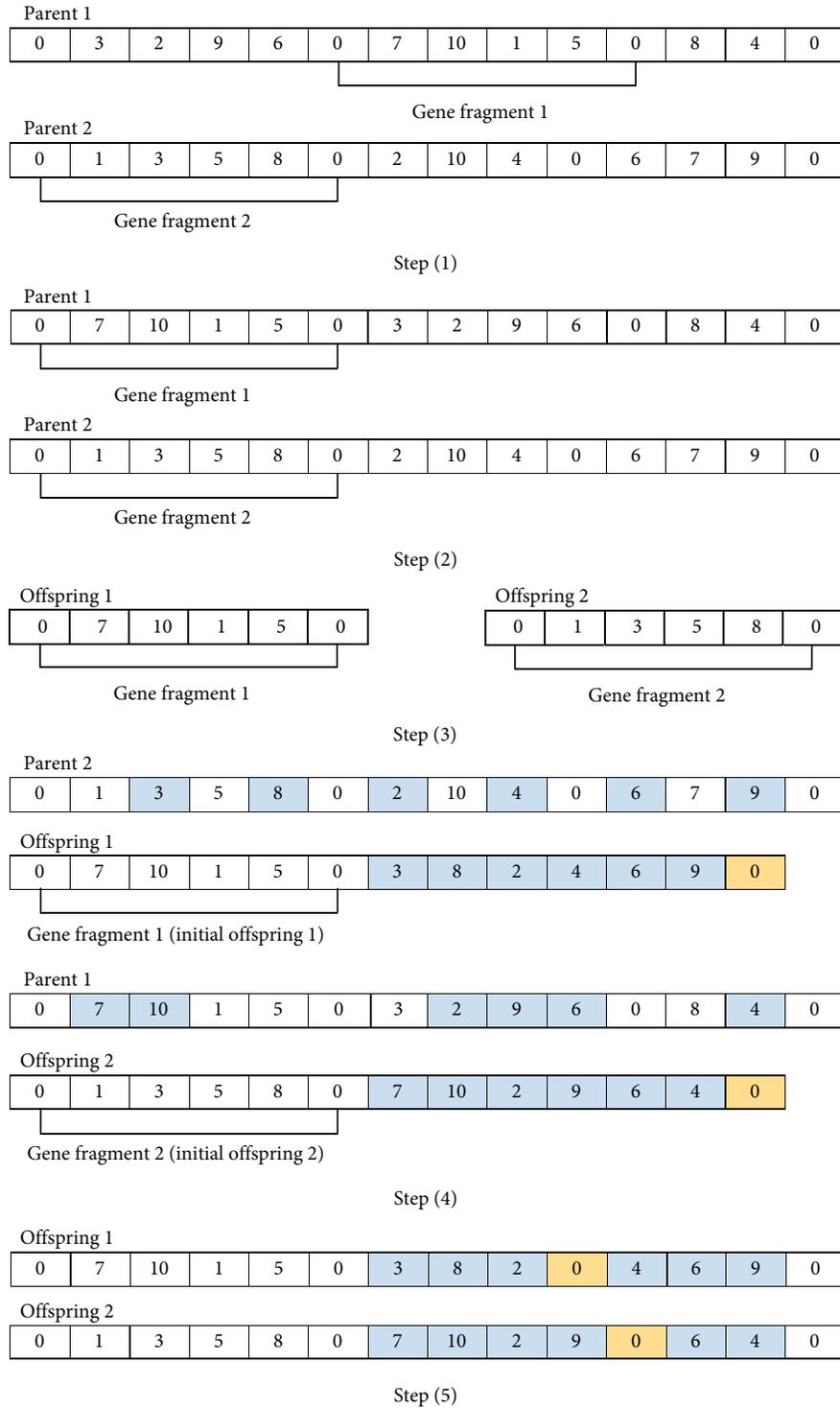


FIGURE 3: The crossover operation when $k > 1$.

After the corresponding divisive areas are obtained, the areas covered by clusters, which are divided from the original data set, are treated as rebalancing units on the top level. The renting/returning number of bikes in each cluster is then counted. Based on the statistic result, a corresponding rebalancing scheme is formulated, which is also the rebalancing strategy on the top level. Then, the subclusters of the

bike stations on the next level are obtained for each rebalancing unit. The clusters at this level are more refined than the previous ones. The corresponding rebalancing units are formed according to the range of each cluster. These units are treated as rebalancing cells on the second level. Thereafter, all clusters on different levels are traversed according to such a rule. The corresponding rebalancing scheme is

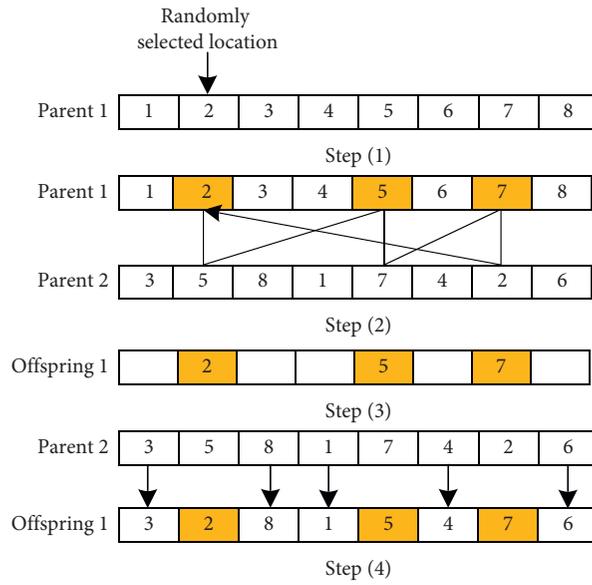


FIGURE 4: The crossover operation when $k = 1$.

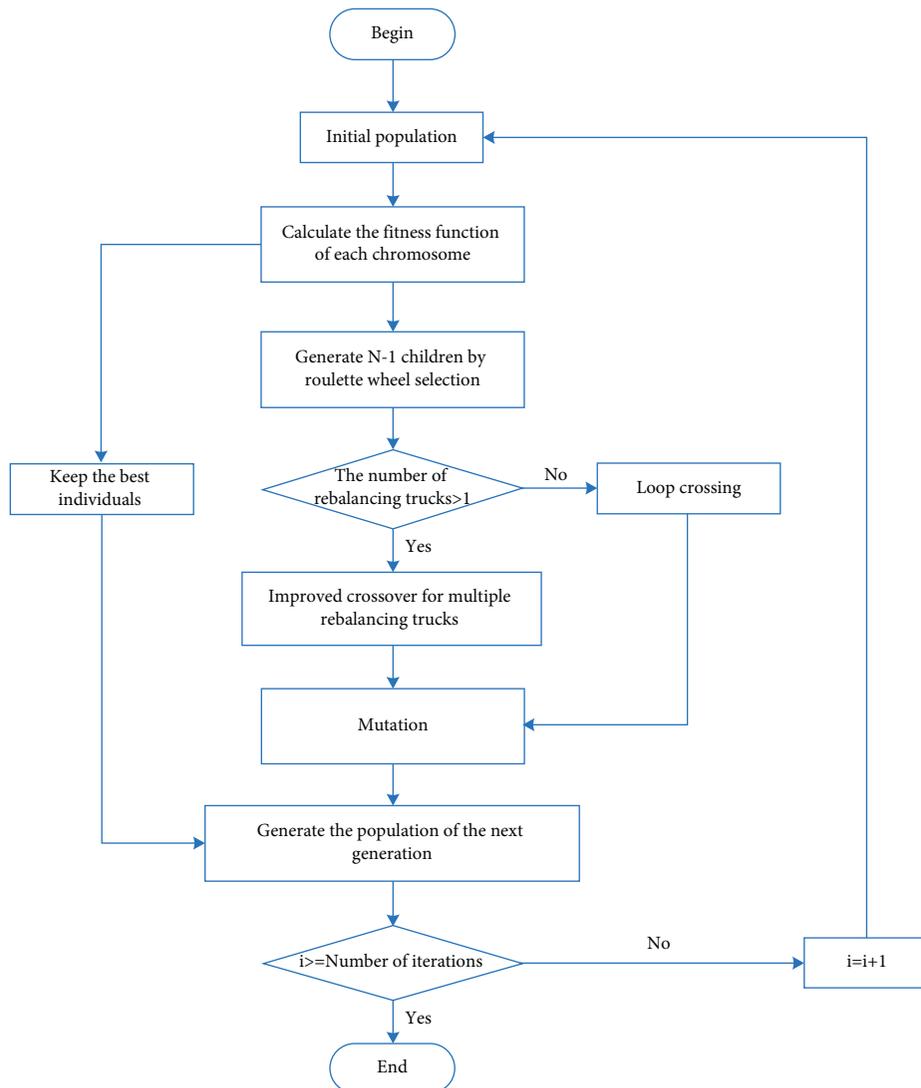


FIGURE 5: Flow of the genetic algorithm.

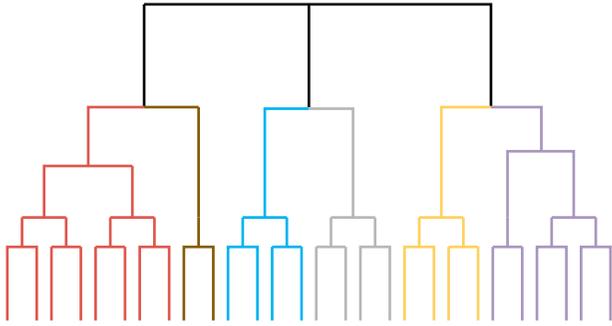


FIGURE 6: K-divisive hierarchical clustering algorithm.

computed according to the renting/returning demand among different clusters on the same level. In this way, the rebalancing strategies on each layer are integrated together to form a tree-like scheme, which is the rebalancing algorithm based on the model of level of detail. The complete process is shown in Figure 7.

In summary, the proposed algorithm is actually a rebalancing process from coarse to fine. This notion also fully reflects the cognitive process of humans from the macro to the micro. When decision-makers are conducting rebalancing schemes, they always carry out the scheme from a macro perspective and then gradually refine it to small ones, rather than focusing on the needs of one or two stations from the beginning. Therefore, the rebalancing algorithm proposed in this work also conforms to the processes and characteristics of cognition and thinking.

4. Results and Discussion

The experiments were conducted using Python, and the program was tested and run under the operating system of Windows 10. Besides, a comparison experiment was performed with the traditional scheme. The results showed that the rebalancing algorithm proposed in this work had obvious advantages compared with the traditional one.

4.1. Experiment for K-Divisive Hierarchical Clustering. The bike-sharing system in the downtown area of Ningbo City is selected as the experimental area in this work. Statistics shows that 169 bike stations are positioned in the area. The area where these stations located is divided into multiple levels of detail, as shown in Figure 8, where one dot represents a bike station. Specifically, the collection of all the stations was regarded as a cluster, which was also treated as the start of the hierarchical division. These stations were used to draw an elbow diagram. A suitable k value was found to divide the cluster that composed of all stations to form small subclusters. These subclusters were regarded as cells on the top level, as areas 1, 2, and 3 shown in Figure 8.

The entire area where all the bike stations locate is divided into three subclusters. These clusters generally cover a large space. The number of bike stations included is relatively large. These stations have a low-density distribution, and the arrangement between stations is relatively sparse. Thereafter, the elbow diagram is drawn again to find the

appropriate k value in each subcluster. The K-means clustering algorithm is applied to further divide the subcluster to obtain a second-level division scheme (e.g., areas 11, 12, and 13 in Figure 8).

The subcluster range will gradually diminish with continuous division. The number of bike stations in each subcluster will also decrease. The division process is continuously performed according to such rules until each cluster contains a small number or only one station. A hierarchical division scheme for stations in bike-sharing system with granularity ranging from large to small is finally formed by overlaying together multiple layers of clusters with different extensions.

4.2. Experiment for Rebalancing Algorithm Based on the Model of LoD. To minimize the amount of calculation, the clusters formed by the stations on the bottom in the division scheme of LoD are treated as the rebalancing units on the last level. In other words, the stations in these cells are no longer to be divided. Combined with the renting/returning demand in each cell, the rebalancing scheme on each layer is computed. Then the rebalancing schemes at each layer are superimposed together to form a complete scheme with granularity from coarse to fine.

The records in rush hours in the morning on June 30, 2019, were selected as manifested in the sample data in the study experiment. The number of bikes at each station at 7:00 a.m. was treated as the initial state. The usage of the bikes at each station was simulated by reading the rental/returning records in the database. The alarm thresholds at stations were set to $[0.4, 0.6]$. Specifically, the station would alarm when its saturation exceeds the threshold interval. Thereafter, a fixed time window starting from the alarming time was generated. The total demand for bikes in each area and its corresponding time window were determined separately, as shown in Table 1.

The genetic algorithm was applied to compute the optimal path among rebalancing units. The times that trucks return to the rebalancing center should be minimized as much as possible. After the rebalancing task at the current level was completed, rebalance for units on the lower level was further performed. Our genetic algorithm supposes the following: the population size is 100, the number of iterations is 1000, the mutation rate is 0.1, the running speed of the truck is 30 km/h, and the maximum carrying capacity of the truck is 400 with an initial load of 250. Finally, the rebalancing schemes at different levels were solved, as shown in Table 2.

4.3. Comparative Experiment and Analysis. We also compared the proposed algorithm with the traditional one. In the traditional algorithm, we did not divide the area where the station locates with the model of LoD. Instead, the genetic algorithm is directly called to rebalance the areas on the lowest layer in the model of LoD of the clusters.

In this experiment, we only assigned one truck to rebalance the bike-sharing system to clarify the results. It is supposed that the running speed of the truck is 30 km/h. The

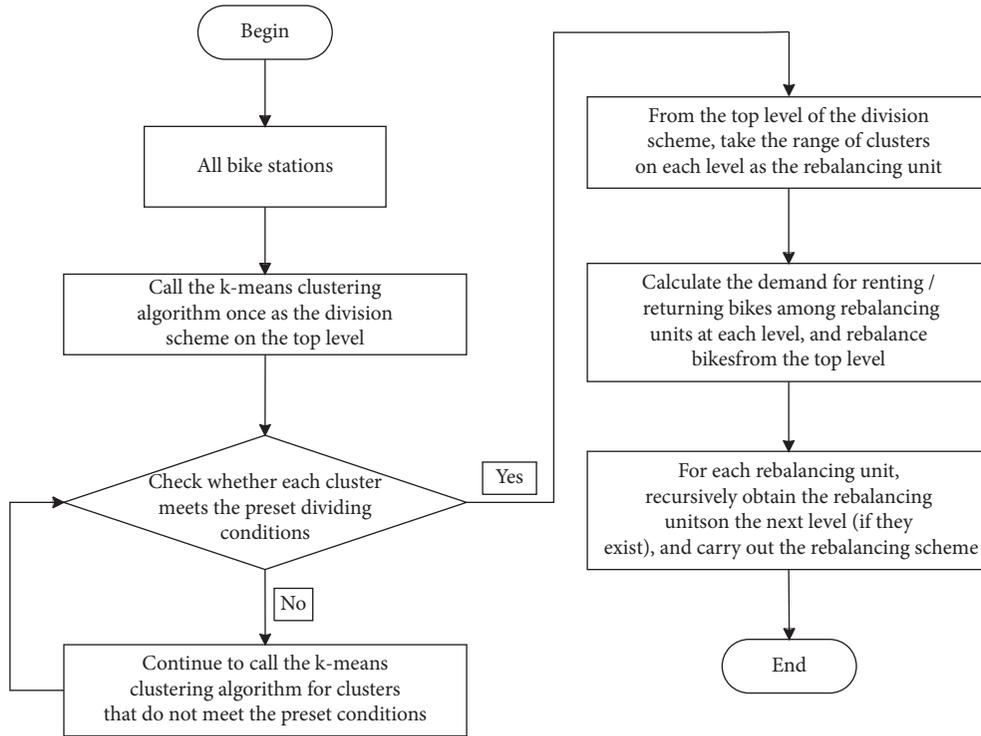


FIGURE 7: Flowchart of the proposed algorithm.



FIGURE 8: Hierarchical division scheme for bike stations in the downtown area of Ningbo.

TABLE 1: Demands for bikes and the corresponding time windows in each rebalancing unit.

Area number	Number of piles	Demand	Time window
C1	1702	170	7:46--8:46
C2	1696	-134	8:16--9:16
C3	1341	-70	8:20--9:20
C11	697	-35	8:15--9:15
C12	340	67	7:53--8:53
C13	665	38	7:43--8:43
C21	383	41	7:45--8:45
C22	405	46	7:43--8:43
C23	450	46	8:07--9:07
C24	458	-29	7:48--8:48
C31	280	39	8:05--9:05
C32	391	-39	8:24--9:24
C33	385	29	8:05--9:05
C34	285	170	7:46--8:46
C111	125	-13	7:52--8:52
C112	120	13	7:41--8:41
C113	192	20	7:50--8:51
C115	115	-12	7:50--8:50
C121	85	-9	8:01--9:01
C123	130	14	8:18--9:18
C132	105	11	7:51--8:51
C133	85	9	7:46--8:46
C134	155	16	7:43--8:43
C135	130	14	7:52--8:52
C211	75	8	7:48--8:48
C212	113	11	7:42--8:42
C213	195	20	7:42--8:42
C221	150	16	7:44--8:44
C222	130	-14	7:55--8:55
C223	125	13	7:47--8:47
C231	195	20	7:37--8:37
C242	70	8	7:37--8:37
C244	106	11	7:45--8:45
C311	160	-17	7:51--8:51
C312	120	-13	7:43--8:43
C321	140	-15	7:41--8:41
C322	115	12	7:38--8:38
C323	136	-14	7:45--8:45
C331	210	-22	7:49--8:49
C332	175	18	7:51--8:51
C341	55	6	7:39--8:39
C342	55	6	7:37--8:37
C343	175	18	8:01--9:01

experimental results are shown in Figure 9. Figure 9(a) shows that the truck would directly run towards all rebalancing units from the center with the traditional rebalancing algorithm. Figure 9(b) shows the rebalancing scheme based on model of LoD proposed in this work.

If the traditional rebalancing method is adopted, then the truck would directly run to all stations from the rebalancing center. The rebalancing scheme for each station was calculated by the genetic algorithm, as shown in Table 3. The total length covered by the truck is 65318.75 m, and the effective time is 130.7 min as shown in Table 4. However, on the basis of the rebalancing scheme proposed in this work, we collected the statistical data on different levels and found that the total length of the path using the proposed method in this paper is 46845.62 m, and the effective time is 93.73 min as shown in

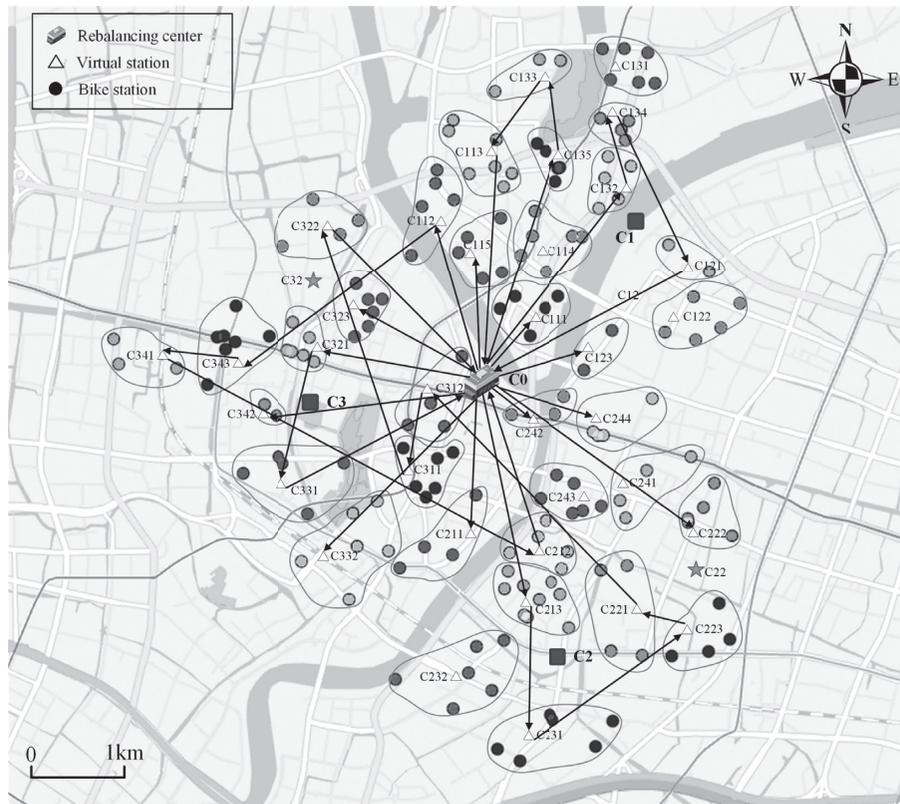
Table 4. It is concluded that the proposed algorithm can reduce the effective time for rebalancing bike-sharing system by 28.3% compared with the traditional one.

The rebalancing method proposed in this work can effectively shorten the total distance, thereby reducing the effective time. This mechanism can help to satisfy the needs of users to use bikes as much as possible and improve the overall service of bike-sharing system in urban areas. Therefore, this rebalancing scheme is practical for bike-sharing system.

The rebalancing algorithm is the key to solving the imbalance distribution of the demand for bikes in the space and time. The experiment proves that the rebalancing method proposed in this work can upgrade the response speed, improve the service of the bike-sharing system, and

TABLE 2: Rebalancing schemes computed by the proposed algorithm.

Rebalancing area	Rebalancing path	Number of bikes that rebalanced	Cumulative distance/m
Original area	C0 → C3 → C2 → C0	304	5845.56
1	C1 → C13 → C11 → C12 → C1	172	4948.04
11	C11 → C111 → C115 → C112 → C113 → C11	58	3266.56
12	C12 → C121 → C123 → C12	23	2298.27
13	C13 → C133 → C135 → C132 → C134 → C13	50	2780.79
2	C2 → C23 → C21 → C24 → C22 → C2	171	5725.94
21	C21 → C213 → C212 → C211 → C21	39	2402.29
22	C22 → C222 → C223 → C221 → C22	43	3086.06
23	C23 → C231 → C23	20	952.48
24	C24 → C242 → C244 → C24	19	1505.56
3	C3 → C34 → C32 → C31 → C33 → C3	136	5740.13
31	C31 → C312 → C311 → C31	30	1201.95
32	C32 → C322 → C323 → C321 → C32	41	2629.86
33	C33 → C332 → C331 → C33	40	1660.51
34	C34 → C341 → C343 → C342 → C34	30	2801.62



(a)

FIGURE 9: Continued.

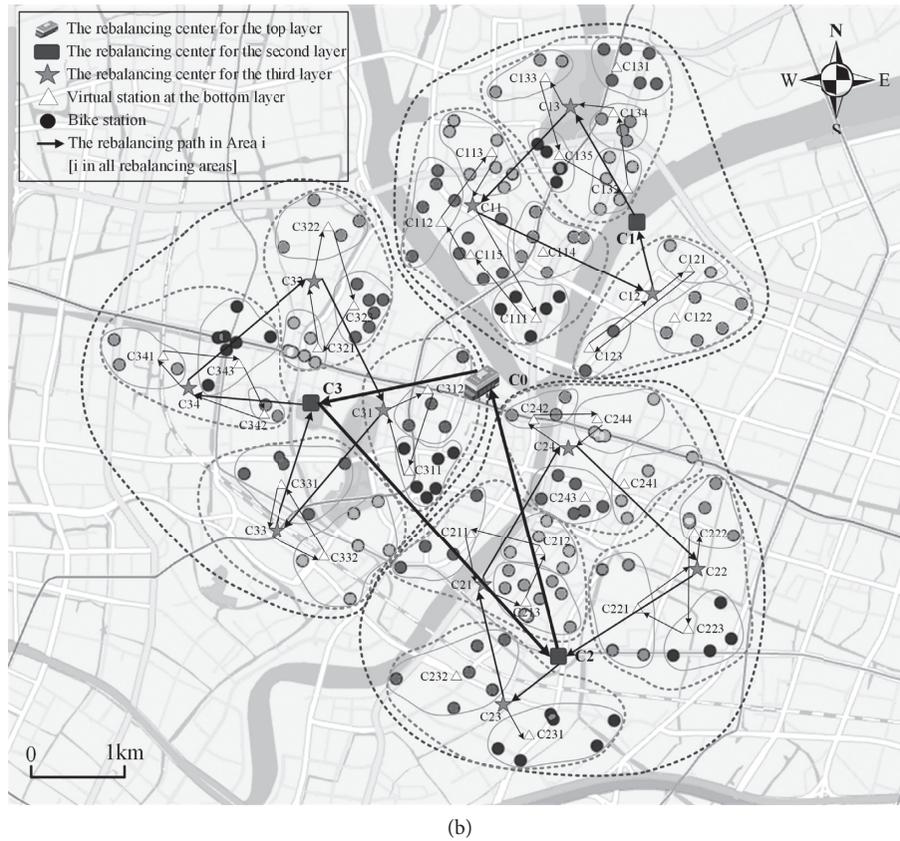


FIGURE 9: Comparison of the traditional rebalancing scheme and the one proposed in this work. (a) Traditional rebalancing scheme. (b) Rebalancing scheme proposed in this work.

TABLE 3: Length of the rebalancing path and time by the traditional method.

Path	Distance (m)	Time consumed (min)
Path 1	5845.10	11.69
Path 2	2124.06	4.25
Path 3	2379.34	4.76
Path 4	1481.96	2.97
Path 5	4789.14	9.58
Path 6	5701.86	11.41
Path 7	900.13	1.80
Path 8	3961.12	7.93
Path 9	1898.28	3.80
Path 10	3679.45	7.36
Path 11	2843.11	5.69
Path 12	2623.59	5.25
Path 13	4244.56	8.49
Path 14	12828.01	25.67
Path 15	10019.04	20.05
Total	65318.75	130.7

TABLE 4: Length of the rebalancing path and time by the proposed method.

Area	Distance (m)	Time consumed (min)
Original area	5845.56	11.70
1	4948.04	9.90
11	3266.56	6.54
12	2298.27	4.60
13	2780.79	5.56
2	5725.94	11.46
21	2402.29	4.81
22	3086.06	6.17
23	952.48	1.91
24	1505.56	3.01
3	5740.13	11.48
31	1201.95	2.40
32	2629.86	5.26
33	1660.51	3.32
34	2801.62	5.61
Total	46845.62	93.73

reduce the maintenance cost. If the method proposed in this work is applied to the operation and management of the bike-sharing system in major cities, then it will not only help improve the service of such system and increase the satisfaction from citizens but also effectively guide people to use bike-sharing system instead of cars. Accordingly, the current situation of traffic congestion on the road is enhanced.

5. Conclusion

We studied a rebalancing scheme for bike-sharing system on the basis of the model of level of detail combined with the K-divisive hierarchy clustering algorithm. The algorithm takes the comprehensive consideration of maximizing to satisfy the renting or returning demand for bikes and minimizing the rebalancing cost as the goals to be optimized.

The algorithm initially treats the collection of all stations as a cluster and then uses the K-means algorithm to cluster them. The obtained subclusters are then reclustered using K-means to form small subclusters. Clustering is continuously executed according to such a rule to form hierarchical layers. The areas where bike stations locate are divided with different granularities. In each layer of the hierarchical structure, the extent covered by the cluster is regarded as a rebalancing unit. Finally, genetic algorithm is called to compute the optimal rebalancing route among units at each level. A rebalancing scheme from coarse to fine is formed from a macro perspective.

Meanwhile, we take bike-sharing system in the downtown area of Ningbo as an example to verify the feasibility and practicability of the algorithm. The comparative experiments proved that the rebalancing scheme can effectively relieve the problem of “difficult to rent/return a bike” in the bike-sharing system. Compared with the traditional method, the algorithm proposed can reduce the effective time for rebalancing bike-sharing system by 28.3%. Therefore, this work can provide an effective theoretical basis for rebalancing bike-sharing system in urban areas and important guidance for management departments.

Although this work has achieved considerable results, some shortcomings are also noted. (1) In establishing the model, the length of the time window is set to a fixed value. We did not discuss the setting of this value in detail. In our future research, the characteristics of demand for renting/returning bikes at stations will be comprehensively considered to analyze the effect of setting different values of time window on the algorithm. This task is necessary to determine the optimal value of the time window. (2) The experiments involved in this study are only used to verify the algorithm’s effectiveness. In the future research, we will combine the spatiotemporal distribution characteristics of renting or returning demand of bikes with a neural network to predict the renting/returning requirements. By doing so, the algorithm proposed could be more accurate.

The research contribution will bring inspiration or guidance to researchers engaged in this field. We will further study the dynamic and intelligent rebalancing strategies for the bike-sharing system in urban areas.

Data Availability

The sample data and codes that support the findings of this study are available at <https://figshare.com/s/b1f7067d0e6ce05bb003>.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was funded by the Natural Science Foundation of Zhejiang Province (no. LQ18D010008), Natural Science Foundation of Ningbo (no. 2018A610132), and a project supported by Scientific Research Fund of Zhejiang Provincial Education Department (Y201736984).

References

- [1] Y. Su, X. Liu, and X. Li, “Research on traffic congestion based on system dynamics: the case of Chongqing, China,” *Complexity*, vol. 2020, 2020.
- [2] H. Luo, “Optimizing bike sharing systems from the life cycle greenhouse gas emissions perspective,” *Transportation Research Part C: Emerging Technologies*, vol. 117, Article ID 102705, 2020.
- [3] J. Qin, S. Lee, X. Yan, and Y. Tan, “Beyond solving the last mile problem: the substitution effects of bike-sharing on a ride-sharing platform,” *Journal of Business Analytics*, vol. 1, no. 1, pp. 13–28, 2018.
- [4] 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.
- [5] B. Legros, “Dynamic repositioning strategy in a bike-sharing system; how to prioritize and how to rebalance a bike station,” *European Journal of Operational Research*, vol. 272, no. 2, pp. 740–753, 2019.
- [6] P. Yi, F. Huang, and J. Peng, “A rebalancing strategy for the imbalance problem in bike-sharing systems,” *Energies*, vol. 12, no. 13, p. 2578, 2019.
- [7] J. Dötterl, “Towards dynamic rebalancing of bike sharing systems: an event-driven agents approach,” in *EPIA Conference on Artificial Intelligence*, Springer, Berlin, Germany, September 2017.
- [8] S. J. Patel and C. R. Patel, *An Infrastructure Review of Public Bicycle Sharing System (PBSS): Global and Indian Scenario*, Springer Singapore, Singapore, 2019.
- [9] J. X. Cao, C. C. Xue, M. Y. Jian, and X. R. Yao, “Research on the station location problem for public bicycle systems under dynamic demand,” *Computers & Industrial Engineering*, 127, pp. 971–980, 2019.
- [10] Y. Zhu and M. Diao, “Understanding the spatiotemporal patterns of public bicycle usage: a case study of Hangzhou, China,” *International Journal of Sustainable Transportation*, vol. 14, no. 3, pp. 163–176, 2020.
- [11] Z. Tian, J. Zhou, W. Y. Szeto, L. Tian, and W. Zhang, “The rebalancing of bike-sharing system under flow-type task window,” *Transportation Research Part C: Emerging Technologies*, vol. 112, pp. 1–27, 2020.
- [12] G. Xu, “A mixed rebalancing strategy in bike sharing systems,” *Engineering Optimization*, vol. 54, pp. 1–18, 2021.
- [13] J. Sultan, G. Ben-Haim, J.-H. Haunert, and S. Dalyot, “Extracting spatial patterns in bicycle routes from crowd-sourced data,” *Transactions in GIS*, vol. 21, no. 6, pp. 1321–1340, 2017.
- [14] C. Yang and G. Gidófalvi, “Mining and visual exploration of closed contiguous sequential patterns in trajectories,” *International Journal of Geographical Information Science*, vol. 32, no. 7, pp. 1282–1304, 2018.
- [15] C. M. Vallez, M. Castro, and D. Contreras, “Challenges and opportunities in dock-based bike-sharing rebalancing: a systematic review,” *Sustainability*, vol. 13, no. 4, p. 1829, 2021.
- [16] A. Yin, B. Ning, and Y. Wang, “Study on bike sharing rebalancing: evidence from kunming,” in *Resilience and Sustainable Transportation Systems*, pp. 280–289, American Society of Civil Engineers Reston, VA, USA, 2020.
- [17] P. Lu, “Local ratio based distributed bike-dock pair allocation in public bike system,” in *Proceedings of the ACM Turing Celebration Conference-China*, Chengdu, China, May 2019.
- [18] L. Shi, Y. Zhang, W. Rui, and X. Yang, “Study on the bike-sharing inventory rebalancing and vehicle routing for bike-

- sharing system,” *Transportation research procedia*, vol. 39, pp. 624–633, 2019.
- [19] Z. Liu and L. Ren, “A sharing bike scheduling optimization algorithm based on two-dimensional dynamic model and improved genetic algorithm,” in *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, IEEE, Halifax, Canada, July 2018.
- [20] D. Mao, “Bike-sharing dynamic scheduling model based on spatio-temporal graph,” in *2018 IEEE International Conference on Big Data and Smart Computing (BigComp)*, IEEE, Bangkok, Thailand, January 2018.
- [21] B. Lahoorpoor, H. Faroqi, A. Sadeghi-Niaraki, and S.-M. Choi, “Spatial cluster-based model for static rebalancing bike sharing problem,” *Sustainability*, vol. 11, no. 11, p. 3205, 2019.
- [22] L. Fan, C. Li, and M. Shi, “Hierarchy training strategy in image classification,” in *2018 Sixth International Conference on Advanced Cloud and Big Data (CBD)*, IEEE, Lanzhou, China, August 2018.
- [23] Y. Wang, “A hierarchical control strategy of microgrids toward reliability enhancement,” in *2018 International Conference on Smart Grid (icSmartGrid)*, IEEE, Nagasaki, Japan, December 2018.
- [24] H. Zhenghua, M. Lingkui, and Z. Wen, “Relational database extension oriented, self-adaptive imagery pyramid model,” *Acta Geodaetica et Cartographica Sinica*, vol. 44, no. 6, p. 678, 2015.
- [25] Z. Hu, T. Jia, G. Wang, J. Wang, and L. Meng, “Computing a hierarchy favored optimal route in a Voronoi-based road network with multiple levels of detail,” *International Journal of Geographical Information Science*, vol. 31, no. 11, pp. 2216–2233, 2017.
- [26] F. K. Nakano, “Top-down strategies for hierarchical classification of transposable elements with neural networks,” in *2017 International Joint Conference on Neural Networks (IJCNN)*, IEEE, Anchorage, AK, USA, May 2017.
- [27] 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.
- [28] W. Li, “Understanding intra-urban human mobility through an exploratory spatiotemporal analysis of bike-sharing trajectories,” *International Journal of Geographical Information Science*, vol. 61, pp. 1–24, 2020.
- [29] P. Zhao, “An empirical study on the intra-urban goods movement patterns using logistics big data,” *International Journal of Geographical Information Science*, vol. 42, pp. 1–28, 2018.
- [30] Y. Xie, “Code similarity detection technique based on AST unsupervised clustering method,” in *2020 IEEE 6th International Conference on Computer and Communications, ICCO*, Chengdu, China, December 2020.
- [31] S. Sivanandam and S. Deepa, “Genetic algorithms,” in *Introduction to Genetic Algorithms*, pp. 15–37, Springer, Berlin, Germany, 2008.
- [32] D. B. Fogel, “Applying evolutionary programming to selected traveling salesman problems,” *Cybernetics & Systems*, vol. 24, no. 1, pp. 27–36, 1993.
- [33] P. K. Amalman and C. F. Eick, *Avalanche: A Hierarchical, Divisive Clustering Algorithm*, Springer International Publishing, Cham, Switzerland, 2015.
- [34] N. Lavrac, “Visual divisive hierarchical clustering using k-means,” 2012.
- [35] M. V. Reddy, M. Vivekananda, and R. Satish, “Divisive hierarchical clustering with K-means and agglomerative hierarchical clustering,” *International Journal of Computer Science Trends and Technology*, vol. 5, no. 5, pp. 6–11, 2017.