

Research Article

Optimizing the Location of Virtual Stations in Free-Floating Bike-Sharing Systems with the User Demand during Morning and Evening Rush Hours

Zhuo Sun ^{1,2}, Yiming Li ¹ and Yi Zuo ^{2,3}

¹Transportation Engineering College, Dalian Maritime University, Dalian 116026, China

²Collaborative Innovation Center for Transport Studies, Dalian Maritime University, Dalian 116026, China

³Navigation College, Dalian Maritime University, Dalian 116026, China

Correspondence should be addressed to Yi Zuo; zuo@dmlu.edu.cn

Received 2 August 2018; Revised 16 December 2018; Accepted 6 February 2019; Published 3 March 2019

Academic Editor: Gonçalo Homem de Almeida Correia

Copyright © 2019 Zhuo Sun 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.

In recent years, free-floating bike-sharing systems (FFBSSs) have been considerably developed in China. As there is no requirement to construct bike stations, this system can substantially reduce the cost when compared to the traditional bike-sharing systems. However, FFBSSs have also become a critical cause of parking disorder, especially during the morning and evening rush hours. To address this issue, the local governments stipulated that FFBSSs are required to deploy virtual stations near public transit stations and major establishments. Therefore, the location assignment of virtual stations is sufficiently considered in the FFBSSs, which is required to solve the parking disorder and satisfy the user demand, simultaneously. The purpose of this study is to optimize the location assignment of virtual stations that can meet the growing demand of users by analyzing the usage data of their shared bikes. This optimization problem is generally formulated as a mixed-integer linear programming (MILP) model to maximize the user demand. As an alternative solution, this article proposes a clustering algorithm, which can solve this problem in real time. The experimental results demonstrate that the MILP model and the proposed method are superior to the K-means method. Our method not only provides a solution for maximizing the user demand but also gives an optimized design scheme of the FFBSSs that represents the characteristics of virtual stations.

1. Introduction

Bike-sharing systems (BSSs) offering a mobility service with public bikes available for shared use are becoming popular in urban environments. With growing awareness of green transportation, the BSSs provide alternate and sustainable carbon-free mode of transportation (especially for short-distance trips) to support a green growth in urban environments and to significantly reduce traffic congestion, pollution, and noise. BSSs permit the travelers to rent a bike at stations and then return it to any station with vacant lockers. Several studies have been widely conducted to optimize the station's location [1], design the shared bikes network [2, 3], and maximize the capacity levels [4].

In recent years, an innovative system for the management of bike-sharing, called free-floating BSSs (FFBSSs), is

gradually developing as an emerging technology [5]. This new system can avoid the necessity of docking stations and kiosk machines with relevant physical and information communication technology infrastructures [6]. Two recent studies have demonstrated the advances of FFBSSs. Caggiani et al. [7] proposed a novel method for the strategic design of FFBSSs whose facilities could be allocated in the territory according to spatial and social equity principles. Leonardo et al. [5] proposed a dynamic and operator-based bike redistribution method that could pursue a decision-making process for the relocation of FFBSSs operating area by predicting the number and position of shared bikes.

In the traditional BSSs (see Figure 1), if people want to go to the bus station from their homes in the morning rush hours, they can unlock a shared bike in the bike station near their home and ride it to the bus station. They can use a

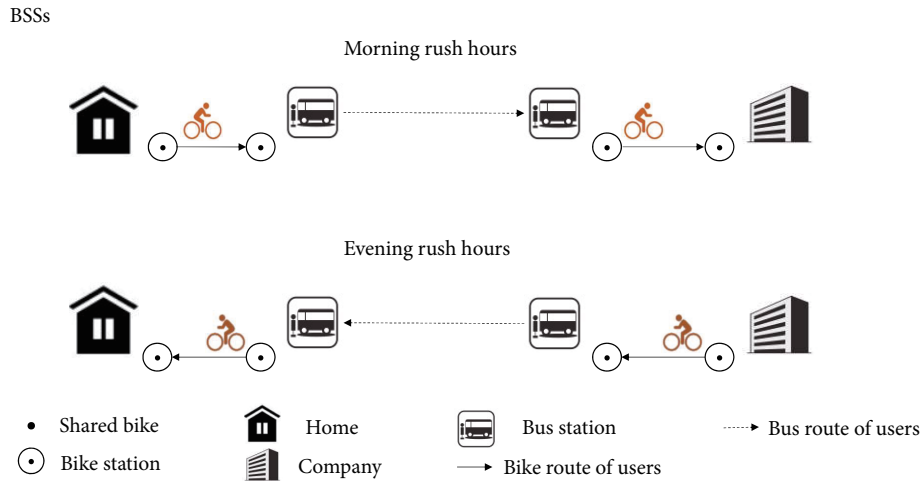


FIGURE 1: Overview of the BSSs.

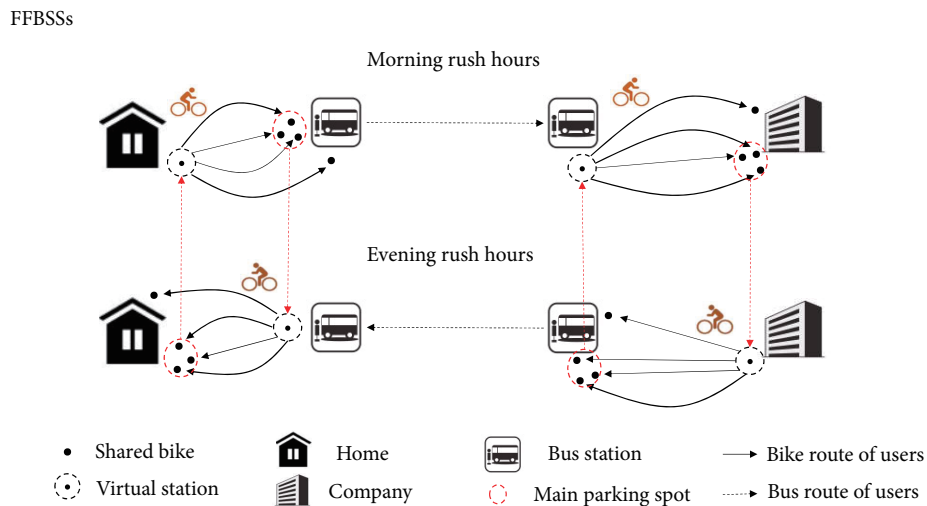


FIGURE 2: Overview of the FFBSSs.

smart phone to find the location of a bike station besides the bus station; however, the users may face the following two problems: there is no shared bike in the bike station near their homes or there is no parking space in the bike station near the bus station. In either case, they have to find another bike station, which would be very inconvenient and waste a lot of time. Conversely, the BSSs' companies need to build bike stations near the home, bus station, and company, which would dramatically increase the maintenance cost of the company. In contrast, FFBSSs can solve such problems as shown in Figure 2. In the morning rush hours, people can find a shared bike in any virtual station near their home and then park it anywhere near the bus station and company. They are not required to be concerned about empty spaces in the virtual station. As the virtual stations are not physical infrastructures, the FFBSSs' companies do not need to build and maintain bike stations. This would sufficiently decrease the cost to the company. Furthermore, FFBSSs also have many advantages such as dynamic adjustment and easy management.

However, though the FFBSSs are superior to the traditional BSSs, it is still a challenging task to decide the location of virtual stations to maximize the satisfaction of user requirements. To address this issue, this study assumes that the people traveling during the morning rush hours use a similar traffic route when they return during the evening rush hours. They expect to find a shared bike in the same place where they had parked it during morning rush hours. The main parking spot in Figure 2 represents the bike parking place of the majority of people. If the bike parking place of most people changes, the main parking spot will also change. As illustrated in Figure 2, during the morning rush hours, people obtain a shared bike from the virtual station and ride it from their homes to the bus station or from the bus station to their companies. Most people would park the shared bikes at the main parking spot, where there is a high user demand during the evening rush hours because most people expect to find a shared bike at the same location during the evening rush hours. The main parking spot of the morning rush hours will probably become the location

of virtual stations of FFBSs during the evening rush hours. Similarly, the main parking spot of evening rush hours has high user demand during the morning rush hours the following day. Therefore, the main parking spot of evening rush hours will probably become the location of virtual stations of FFBSs during morning rush hours the following day. Therefore, optimizing the location of virtual stations while maximizing user demand during morning and evening rush hours is required by both the FFBSs' companies and local governments.

The problem can be described as a mixed-integer linear programming (MILP) model to maximize user demand, and we can use CPLEX and the clustering algorithm [8–10] to obtain the solution. As this problem is a classic nondeterministic polynomial-time hard problem, the computational time will rise exponentially according to data size increments. CPLEX cannot obtain the exact solution in real time, whereas the clustering algorithm can find an approximate solution value in finite iterations. Compared with other methods, the proposed clustering algorithm can not only obtain the characteristics of virtual stations of FFBSs but can also outperform K-means method [11–15].

The rest of this paper is organized as follows. Section 2 reviews numerous studies related to optimizing the location of shared bike stations. Section 3 describes and formulates the proposed problem. Section 4 presents our clustering algorithm. Section 5 depicts the numerical examples. Finally, Section 6 gives our conclusion.

2. Literature Review

The station location of BSSs was studied for decades. Several researches have investigated the optimal location of deploying stations [1], the network design of bike lanes [2, 16], and the maximization of capacity levels [4, 17]. Stations location of BSSs is a strategic decision that depends on their preliminary goals. Location can be efficiently decided by the support of an optimization model, called facility location model [18, 19]. This model considered various objectives, e.g., the minimization of the overall costs and transportation costs and the maximization of demand coverage. Lin and Yang [2] proposed an integer nonlinear program to determine the optimal location of stations. However, the purpose of this work was to minimize cost. The authors did not consider the relocation of bicycles. Martinez et al. [1] employed a mixed-integer linear program to optimize the location of shared bike stations through a heuristic process. The main purpose of this study was to maximize revenue. Romero et al. [17] depicted a bilevel mathematical programming model to optimize the location of public bicycle stations. This paper considered a simulation optimization method that related public bicycles to private cars. Raviv et al. [20] proposed an inventory model to define the management of bike-sharing by depicting a user dissatisfaction function to assess the quality of the relocation service. This model identified the initial inventory of the station so as to minimize the dissatisfaction function. Church and ReVelle [21] also introduced the maximal covering models to maximize the demand coverage. These models and their applications were

widely used for determining the location of BSSs stations based on the maximization of covered demand [22].

In recent years, FFBSs were gradually developed instead of BSSs [5]. There are only two state-of-the-art papers related to the location of virtual stations of FFBSs. Caggiani et al. [7] proposed a strategic designing methodology of FFBSs whose facilities could be allocated in the territory according to spatial and social equity principles. Leonardo et al. [5] proposed a dynamic and operator-based bike redistribution methodology that starts from the prediction of the number and position of bikes over an FFBSs operating area and ends with a decision support system for the relocation process. Therefore, the key to solve the problem of the location of FFBSs' virtual stations is to create an optimization model that considers the maximization of user demand and characteristics of FFBSs' virtual stations. In this study, the location of virtual stations was formulated as a MILP model. We also used a clustering algorithm to solve it. In the field of optimizing location of FFBSs virtual stations, the user demand during morning and evening rush hours has received limited attention. To our knowledge, there is no research that has discussed the location of virtual stations while taking the effect of user demand during morning and evening rush hours into account.

3. Problem Description and Formulation

3.1. Problem Description. In this section, we explain the use of a MILP model to describe the optimizing location of FFBSs' virtual stations. The shared bikes of FFBSs could be parked without the physical stations. However, we could still obtain the parameters of the model such as F_{ij} , Y_i , and X_{ij} by analyzing the usage data of the shared bikes provided by the FFBSs' companies. The model solved the optimal design scheme of virtual stations of FFBSs under the condition of maximization of user demand. We represented the user demand through the number of shared bikes of all virtual stations. The model considered that virtual stations could not exist in isolation. Within a certain distance, each virtual station had a minimum of one virtual station as a support to form a mesh structure. We used the concept of adjacent virtual stations to represent the mutual support between certain virtual stations. We used E_{ij} and L_{ij} to present the lower bound and upper bound of distance between adjacent virtual stations i and j . E_{ij} was also the lower bound of distance of all virtual stations. When the distance of certain virtual stations was in $[E_{ij}, L_{ij}]$, it could be considered that these virtual stations were mutually supportive. When the virtual candidate station was i and j was the adjacent virtual candidate station, then F_{ij} was equal to one. Otherwise, F_{ij} was equal to zero. Owing to the limited area, the quantity of shared bikes in each virtual station was limited.

Figure 4 shows the location of ten shared bikes. Each location could be considered as the location of virtual candidate stations of FFBSs. We used the concept of service radius to represent service area of virtual candidate station. We assumed that shared bike No. 1 was the virtual candidate station. The virtual candidate station No. 1 covered shared bike No. 3 in the service radius. Therefore, $Y_1 = 1$.

3.2. *Formulation.* The problem is formulated using the following notation:

Sets/Indices

N: Set of all virtual candidate stations

Parameters

E_{ij} : Lower bound of distance between adjacent virtual station i and j

F_{ij} : The adjacency matrix of adjacent virtual candidate station i and j

Y_i : The number of bikes of virtual candidate station i

X_{ij} : The distance between virtual candidate stations i and j

H_i : Maximum capacity of virtual station i

S_i : Minimum capacity of virtual station i

K: The number of virtual stations

M: A large positive number

Decision Variables

$Z_i = 1$, if virtual candidate station i is virtual station;
 $Z_i = 0$ otherwise.

Formulation

$$\max \sum_{i \in N} Y_i \cdot Z_i \quad (1)$$

$$\text{s.t.} \quad \sum_{i \in N} Z_i = K \quad (2)$$

$$M \cdot (1 - Z_i) + \sum_{j \in N, j \neq i} F_{ij} \cdot Z_j \geq 1 \quad \forall i \in N \quad (3)$$

$$M \cdot (1 - Z_i) + M \cdot (1 - Z_j) + X_{ij} \cdot Z_j \geq E_{ij} \quad \forall i \in N, \forall j \in N, i \neq j \quad (4)$$

$$Y_i \cdot Z_i \leq H_i \quad \forall i \in N \quad (5)$$

$$M \cdot (1 - Z_i) + Y_i \cdot Z_i \geq S_i \quad \forall i \in N \quad (6)$$

$$Z_i \in \{0, 1\} \quad \forall i \in N \quad (7)$$

The objective function (1) of this linear program maximizes the user demand by maximizing the number of bikes of virtual stations. Constraint (2) indicates that K virtual stations are selected from the virtual candidate stations. Constraint (3) guarantees that each virtual station has one adjacent virtual station at least. Constraint (4) ensures that the distance between virtual stations i and j is greater than or equal to the lower bound E_{ij} of the distance between adjacent virtual stations. Adjacent virtual stations are virtual stations in the network. E_{ij} also represent the lower bound

of the distance of all virtual stations. The number of bikes of the virtual station is lower than maximum capacity of the virtual station according to constraint (5) and higher than a minimum according to constraint (6). Constraint (7) is the binary constraint for the decisional variables.

4. Solution Method

4.1. *Clustering Algorithm.* According to user demand during the morning and evening rush hours, this study proposed a clustering algorithm to optimize the location of virtual stations of FFBSs by analyzing the usage data of shared bikes and considering the characteristics of location of virtual stations of FFBSs. The algorithm randomly generated K centroids and clustered these centroids. We guaranteed that each centroid had a minimum of one adjacent centroid. We calculated the fitness function and selected the optimal solution of the best objective function value. The flowchart is presented in Section 4.2.

The procedure of the clustering algorithm is provided below.

Step 1. Load and process data $D = \{\varphi_1, \varphi_2, \dots, \varphi_m\}$.

Step 2. Set the lower bound E_{ij} and upper bound L_{ij} of distance of adjacent centroid; the number of centroids K; the number of iterations Y; the location of centroid $U = \{U_1, U_2, \dots, U_K\}$; cluster $C = \{C_1, C_2, \dots, C_K\}$; service radius L; Y_i is the number of points in the cluster C_i ; η is the repair coefficient; repair neighborhoods $\psi = \{\psi_1, \psi_2\}$.

Step 3. While the number of iterations $Y \leq Y_{max}$ do.

Step 4. Randomly generate K centroids $U = \{U_1, U_2, \dots, U_k\}$ from data D.

Step 5. Calculate the distance X_{ij} between centroids.

Step 6. If $\forall i \in K, \exists j \in K, i \neq j$ and $X_{ij} \in [E_{ij}, L_{ij}]$, then go to Step 7 else return to Step 4.

Step 7. Calculate the distance d_{ij} between points φ_i and centroids U_j .

Step 8. If $d_{ij} = \|\varphi_i - U_j\| \leq L$ then $C_j = C_j \cup \{\varphi_i\}$.

Step 9. Repair strategy: set η_{max} ; find the $C_{min} = \min_{j=1,2,\dots,K} \{C_j\}$ and repair it with ψ , until $\eta > \eta_{max}$. If the value of C_{min} is better than before, put it back to cluster C. Step 9 is repeated until the value of C_{min} does not change.

Step 10. Calculate the fitness function $Z = \sum_{i=1}^K Y_i$.

Step 11. Compare with the optimal value of history.

Step 12. Repeat Steps 4 –11, until $Y > Y_{max}$.

Step 13. Output the best solution.

4.2. *Flowchart of Clustering Algorithm.* See Figure 3.

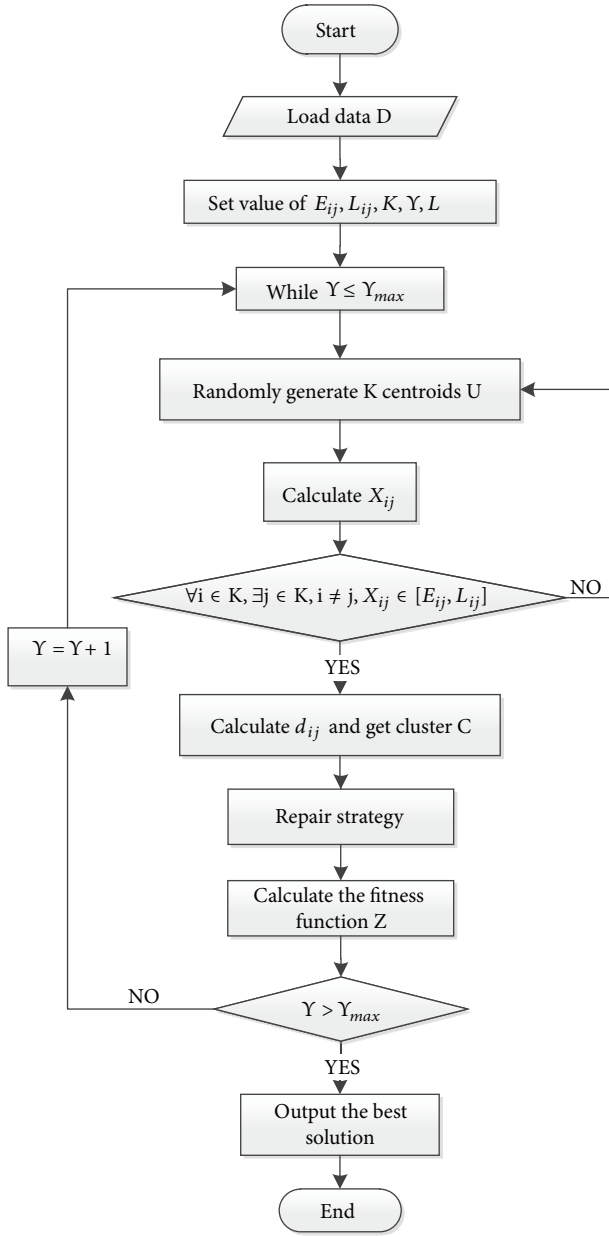


FIGURE 3: Flowchart of clustering algorithm.

5. Numerical Studies

In this section, we describe two numerical examples: a small-scale example and a large-scale example, which are used to illustrate the problem properties and the performance of the clustering algorithm. The small-scale example in Section 5.1 was solved using the branch and cut method in IBM-ILOG CPLEX 12.8 and the clustering algorithm. This example demonstrated the location of 10 shared bikes, and each location could be considered as the location of a virtual candidate station in the FFBS. We selected the location of virtual stations from those virtual candidate stations to maximize user demand. The large-scale example in

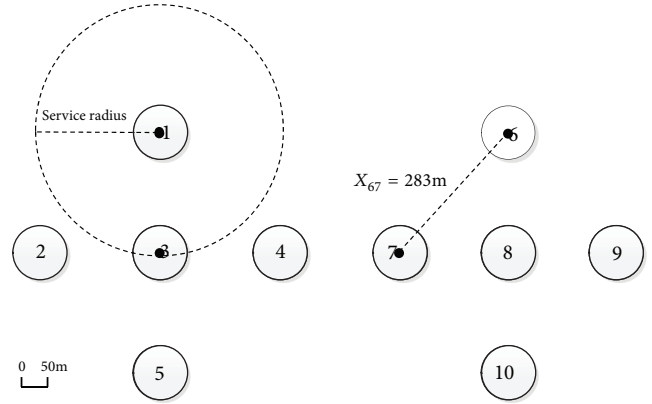


FIGURE 4: Illustration of the network with ten virtual candidate stations.

Section 5.2 was solved with the clustering algorithm to obtain the location of virtual stations of FFBSs in a main area of Beijing in China. The data for the large-scale example comes from a competition of the Mobike algorithm.

5.1. Small-Scale Example. This section makes use of the small network shown in Figure 4 to demonstrate the solution of the model and the clustering algorithm. E_{ij} and L_{ij} are variable parameters. It depends on certain factors, such as terrain and city size. The empirical values of E_{ij} and L_{ij} were obtained from the shared bike company Mobike. A scenario was investigated, and the parameter settings were as follows:

- (1) The lower bound of distance of adjacent virtual stations E_{ij} was set as 400 m;
- (2) The upper bound of distance of adjacent virtual stations L_{ij} was set as 1000 m;
- (3) The number of virtual stations K was set as 2;
- (4) The maximum capacity of virtual station i was set as 10;
- (5) The minimum capacity of virtual station i was set as 1;
- (6) The adjacency matrix of adjacent virtual candidate station F_{ij} was as follows:

$$F_{ij} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (8)$$

TABLE 1: Number of bikes in each virtual candidate station.

Candidate virtual station i	1	2	3	4	5	6	7	8	9	10
Y_i	1	1	4	2	1	1	2	4	1	1

TABLE 2: Computation results of virtual station decision.

Candidate virtual station i	1	2	3	4	5	6	7	8	9	10
Z_i	0	0	1	0	0	0	0	1	0	0

TABLE 3: Performance analysis of CPLEX and the clustering algorithm. $K = 5$.

N	CPLEX				Our method		
	Obj.	CPU(s)	Avg. Obj.	Avg. gap	Best Obj.	Best gap	Avg. CPU(s)
30	15	64.231	14.6	2.67	15	0.00	38.841
50	32	650.289	30.8	3.75	31	3.13	43.142
70	49	2275.631	47.1	3.88	48	2.04	54.260
90	68	7200	65.3	3.97	66	2.94	60.394
110	85	7200	81.5	4.12	82	3.53	69.787
150	114	7200	108.1	5.18	109	4.39	81.140
190	–	7200	149.5	–	153	–	89.769
230	–	7200	176.8	–	190	–	96.145

Avg. Obj: average objective value; Best Obj: best objective value; Avg. CPU: average CPU time after 100 runs.

(7) The matrix of X_{ij} was as follows:

$$X_{ij} = \begin{bmatrix} 0 & 283 & 200 & 283 & 400 & 600 & 447 & 632 & 825 & 721 \\ 283 & 0 & 200 & 400 & 283 & 825 & 600 & 800 & 1000 & 825 \\ 200 & 200 & 0 & 200 & 200 & 632 & 400 & 600 & 800 & 632 \\ 283 & 400 & 200 & 0 & 283 & 447 & 200 & 400 & 600 & 447 \\ 400 & 283 & 200 & 283 & 0 & 721 & 447 & 632 & 825 & 600 \\ 600 & 825 & 632 & 447 & 721 & 0 & 283 & 200 & 283 & 400 \\ 447 & 600 & 400 & 200 & 447 & 283 & 0 & 200 & 400 & 283 \\ 632 & 800 & 600 & 400 & 632 & 200 & 200 & 0 & 200 & 200 \\ 825 & 1000 & 800 & 600 & 825 & 283 & 400 & 200 & 0 & 283 \\ 721 & 825 & 632 & 447 & 600 & 400 & 283 & 200 & 283 & 0 \end{bmatrix} \quad (9)$$

(8) The number of bikes in virtual candidate station Y_i was as Table 1.

Each location of shared bike could be considered as a virtual candidate station. We used the number of shared bikes in the service radius of the virtual candidate station as the number of shared bikes for the virtual candidate stations. For example, we assumed that the service radius of the virtual candidate station was 200 m. There is only shared bike No. 3 in the service radius of the virtual candidate station No. 1. Therefore, the number of shared bikes of virtual candidate station No. 1 was one.

5.1.1. Result. Based on the problem properties and parameter setting in Section 5.1, the problem was solved with the branch and cut method in CPLEX 12.8 and the clustering algorithm.

These two methods obtained the same answer in a short time. CPLEX obtained an optimal solution in less than

1 s, whereas the clustering algorithm took more than 30 s to obtain the same solution. The latter required more time because it could not stop immediately after obtaining an optimal solution but instead could stop only after a predetermined number of iterations (6000 in this case) without further improvement. The results are listed in Table 2. According to Table 2, the virtual candidate station Nos. 3 and 8 are selected as virtual stations of FFBSs. The value of user demand was 8.

5.1.2. Comparison of the Performance of the Exact Method and the Clustering Algorithm. The clustering algorithm was coded in C++ and all computational experiments were performed using a Lenovo notebook with an Intel Core i7-7700HQ CPU with processor base frequency of 2.80 GHz. The small-scale example is a subset of the large-scale example. This set contains instances of 30-230 virtual candidate stations. The number of virtual stations K is set as 5 in Table 3 and 10 in Table 4. The results obtained from the clustering algorithm were compared to those obtained from CPLEX 12.8 with default setting and a maximum running time of 2 h.

As listed in Table 3, Obj denotes the true optimal objective value from CPLEX. CPU is the running time of CPLEX. Table 3 also lists the Avg. Obj and Avg. CPU obtained by the clustering algorithm in 100 runs. Avg. gap (%) represents the deviation of the average objective value from the Obj. Best gap (%) represents the deviation of the best objective value from the Obj.

When $N = 30$, as listed in Table 3, CPLEX obtained an optimal solution in a time marginally over 1 min, whereas the clustering algorithm required less than 40 s to obtain the same solution. Further, Avg. Obj was nearly equal to Obj value. The Avg. gap was 2.67% and Best gap was 0%. When $N = 50$,

TABLE 4: Performance analysis of CPLEX and the clustering algorithm. $K = 10$.

N	CPLEX				Our method			
	Obj.	CPU(s)	Avg. Obj.	Avg. gap	Best Obj.	Best gap	Avg. CPU(s)	
30	20	78.645	19.5	2.50	20	0.00	39.916	
50	37	681.843	35.7	3.51	36	2.70	45.016	
70	55	2664.101	52.2	5.09	53	3.64	56.791	
90	74	7200	69.9	5.54	71	4.05	63.855	
110	92	7200	86.1	6.41	88	4.35	75.420	
150	128	7200	119.4	6.72	122	4.69	86.632	
190	–	7200	163.1	–	165	–	95.341	
230	–	7200	188.7	–	206	–	100.029	

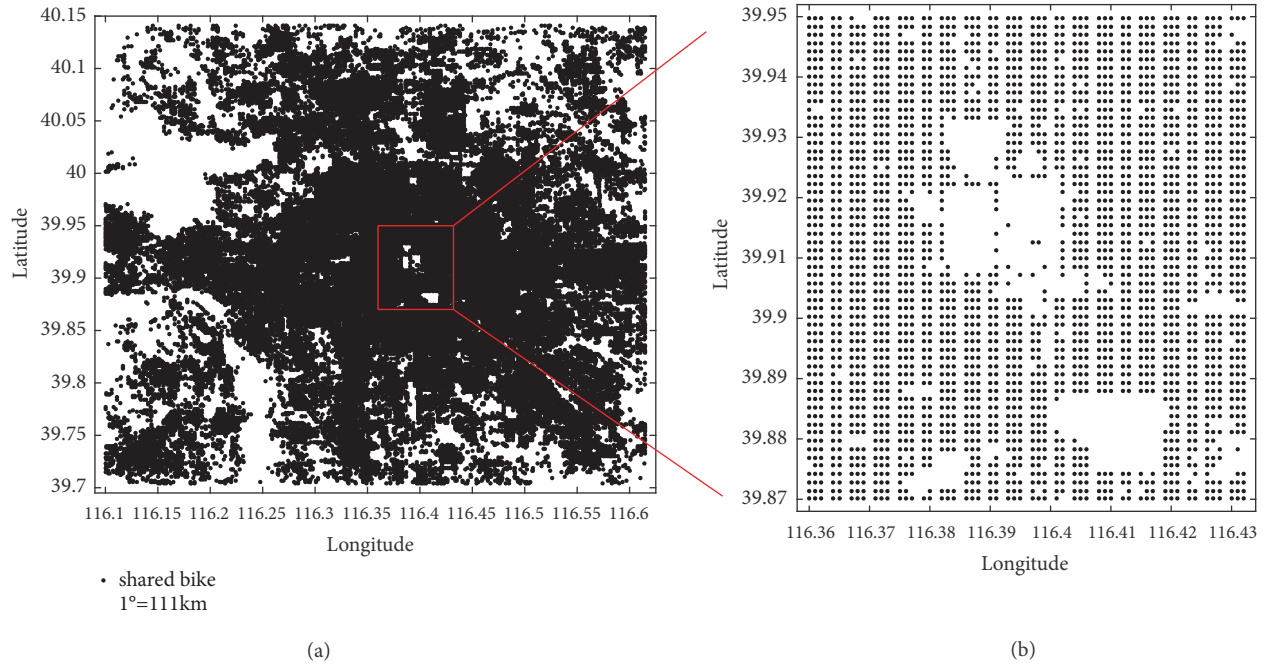


FIGURE 5: Location of shared bikes in the area of: (a) sixth loop of Beijing and (b) second loop of Beijing. The black dots represent the location of shared bikes.

CPLEX achieved an optimal solution in more than 10 min, whereas the clustering algorithm could obtain a good, feasible solution with a Avg. gap of 3.75% and a Best gap of 3.13 in less than 1 min.

For the larger problems ($N = 70, 90, 110, 150$), CPLEX obtained an optimal solution in 2 h. Meanwhile, the clustering algorithm obtained a more feasible solution in only approximately 50-90 s. The Avg. gap and the Best gap obtained by the clustering algorithm increased with the problem sizes, with a Best gap of less than 5% for $N = 150$ in Table 3.

When $N \geq 190$, as listed in Table 3, the clustering algorithm obtained a feasible solution in about 90-100 s, whereas CPLEX was unable to do so in 2 h. This demonstrates the limitations of CPLEX and the strength of our proposed method in large applications.

Table 4 exhibits that when K is set as 10, the value of Obj, Avg. Obj, and Best Obj is better than Table 3. However, it required more time than Table 3. For example, when $N = 30$,

the value of CPU is greater than 70 s in Table 4. However, the CPU is 64.231 s in Table 3. The Avg. gap and the Best gap obtained by the clustering algorithm increased with the problem sizes in Table 4. Tables 3 and 4 demonstrate that when $N \geq 190$, CPLEX could not obtain an optimal solution in 2 h.

To sum up, the clustering algorithm yields a good, feasible solution in shorter running time when $N \geq 30$. Overall, this method produces high-quality solutions with short computing times.

5.2. Large-Scale Example. The large-scale example uses data (<https://biendata.com/competition/mobike/data/>) obtained from a competition of the Mobike algorithm. Figure 5 shows the location of the shared bike that is distributed in the areas of the sixth loop and second loop in Beijing, China. As listed in Table 3, owing to the large-scale problem, CPLEX could not obtain the precise solution in limited time. Therefore, we

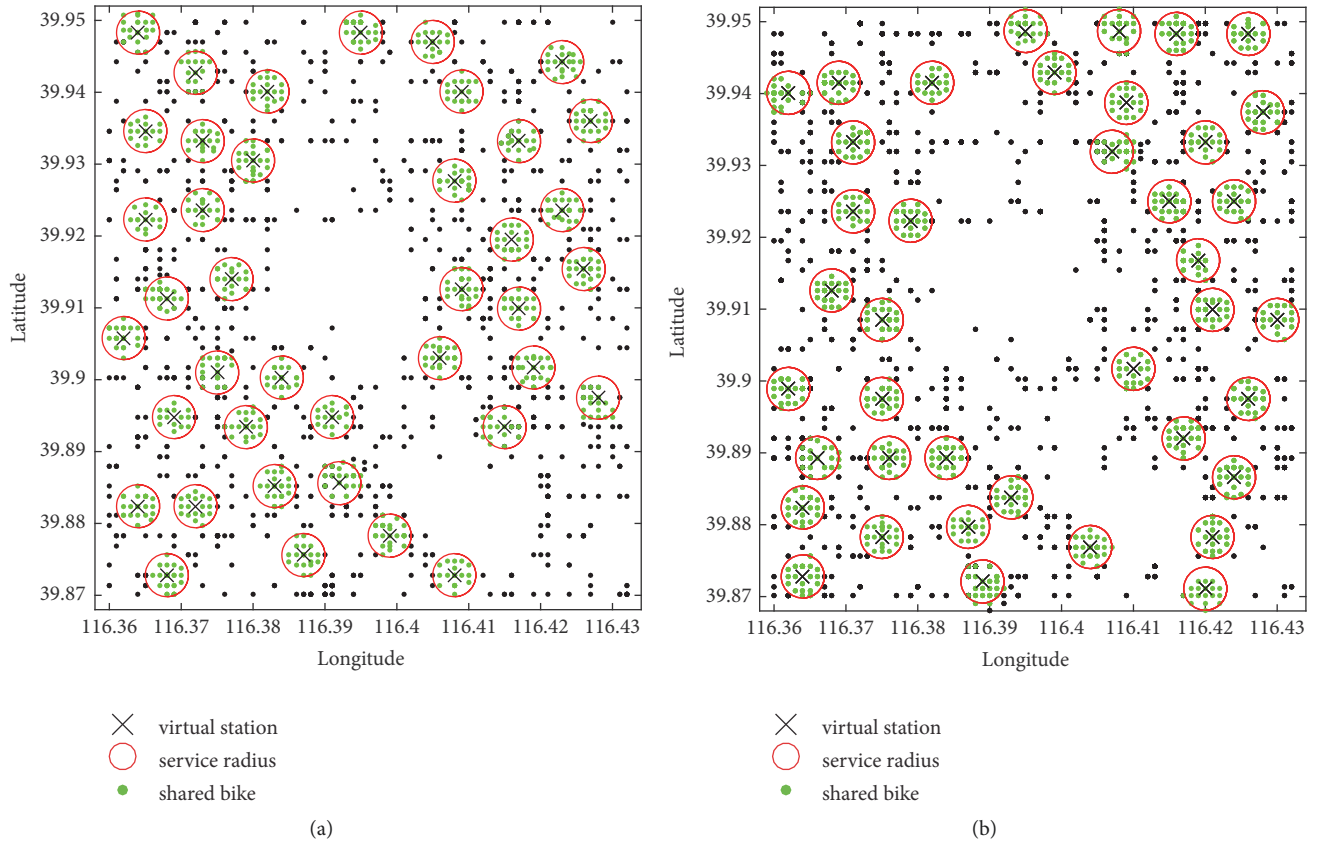


FIGURE 6: Best solution for location of virtual stations of FFBSs during morning and evening rush hours.

used the clustering algorithm to analyze the data of shared bikes used by people at 9 am and 9 pm in the area of second loop of Beijing to obtain the location of virtual stations of FFBSs.

5.2.1. Parameter Setting. This section is composed of the related parameter settings. We will explain the reasoning for setting the quantity of virtual stations to 40 in Section 5.2.4. The service radius of virtual stations and the distance of adjacent virtual stations was derived from relevant data of Mobike. The area of second loop of Beijing was obtained from Google Maps.

- (1) The number of virtual stations K was set as 40;
- (2) The service radius of virtual stations L was set as 200 m;
- (3) The lower bound of distance of adjacent virtual stations E_{ij} was set as 400 m;
- (4) The upper bound of distance of adjacent virtual stations L_{ij} was set as 1000 m;
- (5) The fitness function was the objective function of the model;
- (6) The area of second loop of Beijing: latitude: [39.8698, 39.9497], longitude: [116.3595, 116.4324];
- (7) Y_{max} was 6000;
- (8) The initial objective function value was zero.

5.2.2. Result. Figure 6 shows the best solution for location of virtual stations of FFBSs during morning and evening

rush hours. In this figure, the horizontal axis and vertical axis present the longitude and latitude. The black crosses illustrate the location of virtual stations. The red circles illustrate the service radius of virtual stations. The green dots show the quantity of shared bikes in virtual stations. The black dots indicate shared bikes that are not included in the virtual stations. There are 1585 shared bikes in Figure 6(a) and 1520 shared bikes in Figure 6(b).

We used the clustering algorithm to analyze the usage data of shared bikes at 9 am in the main area of Beijing, to obtain the location where the demand of shared bike is high. To maximize the demand of users, 40 virtual stations were deployed as black crosses in Figure 6(a).

Simultaneously, we also used the same clustering algorithm to analyze the usage data of shared bikes at 9 pm to obtain the location where the demand of shared bike was high. To maximize the demand of users, 40 virtual stations were deployed as black crosses in Figure 6(b), which is different from Figure 6(a). Therefore, the results reveal that FFBSs' virtual stations have dynamic merit that can satisfy the users demand for different rush hours.

5.2.3. Diagram of Algorithm Convergence. In the experiments, we used the clustering algorithm to analyze the usage data of shared bikes at 9 am and 9 pm in the main area of Beijing to obtain the diagram of algorithm convergence. Figure 7 is the diagram of algorithm convergence of Figure 6(a).

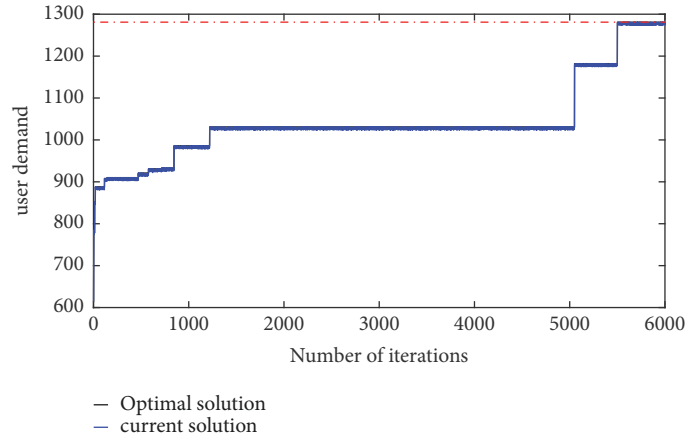


FIGURE 7: Diagram of algorithm convergence of Figure 6(a) of virtual station.

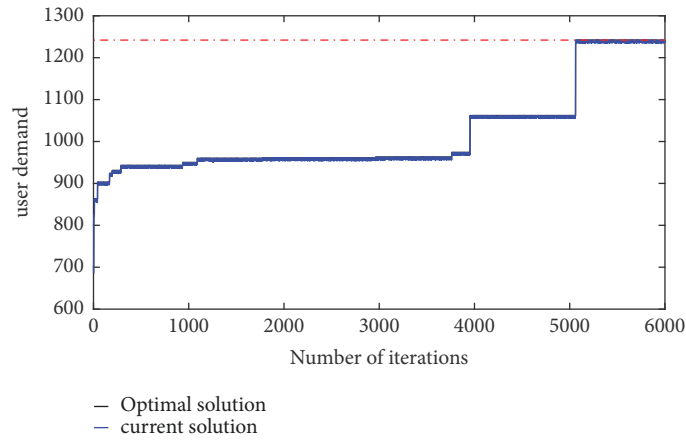


FIGURE 8: Diagram of algorithm convergence of Figure 6(b) of virtual station.

Figure 7 shows the result that the algorithm converges on the 5501st generation and the maximum value is 1281. Figure 8 is the diagram of algorithm convergence of Figure 6(b). Figure 8 illustrates the result that the algorithm converges on 5062nd generation and the maximum value is 1242.

5.2.4. Sensitivity Analysis of Quantity of Virtual Stations. Further, we used the clustering algorithm to estimate the optimal number of virtual stations to obtain the value of user demand in different quantities of virtual stations.

Table 5 depicts user demand increasing owing to the increase in the number of virtual stations. $Avg Y_i$ is the ratio of user demand to virtual station, which also denotes the average user demand for each virtual station. The value of $Avg Y_i$ is maximum when the number of virtual stations is 40. Therefore, we set the number of virtual stations to 40 in the main area of Beijing.

5.2.5. Comparison of the Performance of Different Algorithms. We also compared our method with K-means algorithm. K-means clustering is one of the most popular methods for data clustering and classification. For a large number of highly dimensional data, K-means can provide an efficient approach

to divide similar objects into the same cluster by minimizing the global Euclidean distance. In our numerical experiments, K-means randomly selected the points as initial means, and the means of each cluster was recalculated until it could satisfy the distance requirement of virtual stations of FFBSs.

In contrast, our method selected a series of feasible points as initial solution, which can dramatically enhance the coverage rate of FFBSs' virtual stations. Furthermore, to be compared with the clustering process of K-means method, our method could recluster the objects by maximizing the user demand of each FFBS's virtual station. Table 6 depicts the results of maximizing user demand using our method outperforming that of K-means when the number of virtual stations was 40.

6. Conclusions

This article proposed a method for optimizing the location of virtual stations of FFBSs to achieve the demand of people during morning and evening rush hours. Compared to the fixed stations, the location of virtual stations could change as the user demand changes. Virtual stations were more effective than fixed stations. The problem was solved by a MILP

TABLE 5: Computation results of different quantities of virtual stations.

Computation results of data at 9 a.m.					
Virtual stations	30	35	40	45	50
User demand ($\sum Y_i$)	940	1105	1281	1350	1394
Avg Y_i	31.33	31.57	32.03	30.00	27.88
Computation results of data at 9 p.m.					
Virtual stations	30	35	40	45	50
User demand ($\sum Y_i$)	920	1076	1242	1321	1356
Avg Y_i	30.67	30.74	31.05	29.36	27.12

TABLE 6: Computation results of different algorithm.

Computation results of data at 9 a.m.		
Algorithm	Our method	K-means
User demand ($\sum Y_i$)	1281	995
Computation results of data at 9 p.m.		
Algorithm	Our method	K-means
User demand ($\sum Y_i$)	1242	1001

model and a clustering algorithm based on the maximization of user demand. We further tested our proposed method using data from Mobike. The results demonstrated that our method could effectively obtain the location of FFBSs' virtual stations that could satisfy the user demand during the morning and evening rush hours. We derived the rationality of parameter setting through sensitivity analysis. By comparing the performance of different algorithms, our method was determined to be more effective than the K-means method. According to the diagram of algorithm convergence, our method could converge to a satisfactory value in limited iterations.

This article presents three contributions as follows. First, our method is superior to the CPLEX in CPU time, especially for large data. Second, our method can better consider the characteristics of virtual stations of FFBSs than the existing algorithms. Third, our method exhibits computational efficiency. We could find a satisfactory value in limited iterations. This practical method can be useful for strategic planning in the companies of shared bikes.

In the future, we plan to extend our analysis to consider user demand changes at different times and improve our method to fit this dynamic adjustment of FFBSs' virtual stations. For this purpose, we will also develop our method to cluster objects without fixed cluster numbers.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work is supported in part by the National Natural Science Foundation of China (61304179, 71501021, 71871036, 71431001, U1813203, 71831002, and 71672016), the Science & Technology Innovation Funds of Dalian (2018J11CY022), the Program for Innovative Research Team in University (IRT_17R13), and the Fundamental Research Funds for the Central Universities (3132018301, 3132018304).

References

- [1] L. M. Martinez, L. Caetano, T. Eiró, and F. Cruz, "An optimisation algorithm to establish the location of stations of a mixed fleet biking system: an application to the city of lisbon," *Procedia - Social and Behavioral Sciences*, vol. 54, no. 1, pp. 513–524, 2012.
- [2] J.-R. Lin and T.-H. Yang Ta-Hui, "Strategic design of public bicycle sharing systems with service level constraints," *Transportation Research Part E: Logistics and Transportation Review*, vol. 47, no. 2, pp. 284–294, 2011.
- [3] J. R. Lin, T. H. Yang, and Y. C. Chang, "A hub location inventory model for bicycle sharing system design: formulation and solution," *Computers & Industrial Engineering*, vol. 65, no. 1, pp. 77–86, 2013.
- [4] J. C. García-Palomares, J. Gutiérrez, and M. Latorre, "Optimizing the location of stations in bike-sharing programs: a GIS approach," *Applied Geography*, vol. 35, no. 1-2, pp. 235–246, 2012.
- [5] L. Caggiani, R. Camporeale, M. Ottomanelli, and W. Y. Szeto, "A modeling framework for the dynamic management of free-floating bike-sharing systems," *Transportation Research Part C: Emerging Technologies*, vol. 87, pp. 159–182, 2018.
- [6] A. Pal and Y. Zhang, "Free-floating bike sharing: Solving real-life large-scale static rebalancing problems," *Transportation Research Part C: Emerging Technologies*, vol. 80, pp. 92–116, 2017.
- [7] L. Caggiani, R. Camporeale, and M. Ottomanelli, "Planning and design of equitable free-floating bike-sharing systems

- implementing a road pricing strategy,” *Journal of Advanced Transportation*, vol. 2017, Article ID 3182387, 18 pages, 2017.
- [8] C. Biernacki and G. Govaert, “Choosing models in model-based clustering and discriminant analysis,” *Journal of Statistical Computation and Simulation*, vol. 64, no. 1, pp. 49–71, 1999.
 - [9] C. Fraley and A. E. Raftery, “Model-based clustering, discriminant analysis, and density estimation,” *Journal of the American Statistical Association*, vol. 97, no. 458, pp. 611–631, 2002.
 - [10] G. Galimberti, A. Manisi, and G. Soffritti, “Modelling the role of variables in model-based cluster analysis,” *Statistics & Computing*, pp. 1–25, 2017.
 - [11] S. Z. Selim and M. A. Ismail, “K-means-type algorithms: a generalized convergence theorem and characterization of local optimality,” *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 6, no. 1, pp. 81–87, 1984.
 - [12] D. S. Modha and W. S. Spangler, “Feature weighting on k-means clustering,” *Machine Learning*, vol. 52, no. 3, pp. 217–237, 2003.
 - [13] O. Shamir and N. Tishby, “Stability and model selection in k-means clustering,” *Machine Learning*, vol. 80, no. 2-3, pp. 213–243, 2010.
 - [14] X. Huang, Y. Ye, H. Guo, Y. Cai, H. Zhang, and Y. Li, “Dskmeans: a new kmeans-type approach to discriminative subspace clustering,” *Knowledge-Based Systems*, vol. 70, pp. 293–300, 2014.
 - [15] M. J. Brusco and J. D. Cradit, “A variable-selection heuristic for k-means clustering,” *Psychometrika*, vol. 66, no. 2, pp. 249–270, 2001.
 - [16] G. K. D. Saharidis, A. Fragkogios, and E. Zygouri, “A multi-periodic optimization modeling approach for the establishment of a bike sharing network: a case study of the city of Athens,” in *Proceedings of the International MultiConference of Engineers and Computer Scientists*, vol. 2, no. 2210, pp. 1226–1231, 2014.
 - [17] J. P. Romero, A. Ibeas, J. L. Moura, J. Benavente, and B. Alonso, “A simulation-optimization approach to design efficient systems of bike-sharing,” *Procedia-Social and Behavioral Sciences*, vol. 54, pp. 646–655, 2012.
 - [18] M. S. Daskin, “What you should know about location modeling,” *Naval Research Logistics (NRL)*, vol. 55, no. 4, pp. 283–294, 2008.
 - [19] M. S. Daskin, “Network and discrete location,” *Algorithms and Applications*, vol. 48, pp. 763–763(1), 1995.
 - [20] T. Raviv and O. Kolka, “Optimal inventory management of a bike-sharing station,” *Iie Transactions*, vol. 45, no. 10, pp. 1077–1093, 2013.
 - [21] R. Church and C. ReVelle, “The maximal covering location problem,” *In of the Regional Science Association*, vol. 32, no. 1, pp. 101–118, 1974.
 - [22] I. Frade and A. Ribeiro, “Bike-sharing stations: a maximal covering location approach,” *Transportation Research Part A: Policy and Practice*, vol. 82, pp. 216–227, 2015.



Hindawi

Submit your manuscripts at
www.hindawi.com

