Dynamic Demand-driven Bike Station Clustering

Abstract

As an eco-friendly transportation option, bike-sharing systems have become increasingly popular because of their low costs and contributions to reducing traffic congestion and emissions generated by vehicles. Due to the availability of bikes and the geographically varied bike flows, shared-bike operators have to reposition bikes and perform the required bike loading and unloading activities throughout the day in a large and dynamic shared-bike network. Most of the existing studies cluster bike stations by their geographical locations to form smaller sub-networks for more efficient optimization of bike-repositioning operations. This study develops a new methodological framework with a demand-driven approach to clustering bike stations in bike-sharing systems. Our approach captures spatiotemporal patterns of user demands and can enhance the efficiency of bike-repositioning operations. A directed graph is constructed to represent the bike-sharing system, whose vertices are bike stations and arcs represent bike flows, weighted by the number of trips between the bike stations. A novel demand-driven algorithm based on community detection is developed to solve the problem. Numerical experiments are conducted with the data captured from the world's largest bike-sharing system, consisting of nearly 3,000 stations. The results show that, with CPLEX solutions as the benchmark, the proposed methodology provides high-quality solutions with shorter computing times. The clusters identified by our methodology are effective for bike repositioning in terms of the performance metrics, such as the number of bike stations in each cluster and the geographic proximity. The comparison between clusters found in different hours indicates that bike sharing is a short-distance transportation mode. One of the key conclusions from the computational study is that clustering bike stations by bike flow in the network not only enhances the efficiency of bike-repositioning operations but also preserves the geographic characteristics of clusters.

Keywords: Demand-driven clustering, bike sharing, bike repositioning, community detection

1 Introduction

Bike-sharing systems (BSSs) have increasingly emerged to fill gaps in today's transportation networks. BSSs provide an alternative and lower-cost solution for short-distance transportation. These systems contribute to reducing traffic congestion and pollution caused by motorized transportation. As one of the largest BSSs in the world, CitiBike (New York) decreased more than 2.9 million pounds of carbon emissions in June 2018 (Negahban, 2019). Due to the lower travel cost and environmental benefits of BSSs, as of 2020, there are more than 2,000 bike sharing systems around the world and approximately 17,792,000 bikes in

service (Shui and Szeto, 2020).

To attract more users, many bike sharing companies focus on scale expansion rather than profits, and release large quantities of shared bikes in a short time (Hasija et al., 2020; Sun et al., 2020). The number of bikes ranges from less than one hundred to many thousands in some cities like Hangzhou (78,000 bikes and 4,198 bike stations) and Paris (20,600 bikes and 1,338 bike stations) (Alvarez-Valdes et al., 2016). BSSs can be categorized into two classes: station-based BSSs (SBBSSs) and free-floating BSSs (FFBSSs). In a SBBSS, users can pick up and drop off bikes only at fixed-location bike stations. Such a network can be depicted by a graph with nodes representing bike stations. On the other hand, in FFBSSs, bikes can be picked up and dropped off anywhere. In some studies, nodes are also considered in FFBSSs, where these nodes include regular bike docks and locations where customers rent or return bikes (Pal and Zhang, 2017). In this study, we focus on SBBSSs and construct a graph to represent the bike station network (BSN).

The success of a bike-sharing system depends on its ability to ensure the availability of bikes and parking spaces at the right bike stations at the right times (Albiński et al., 2018). However, due to the imbalanced bike flows during the day, the availability of bikes and user demands are often mismatched (Dell'Amico et al., 2018). For example, many users may choose to ride bikes from a bus station to a residential area after work. Subsequently, bikes originally available at the bus station would be retrieved and returned in the residential area. Future users may, therefore, not be able to access the bikes. To maintain a good service level, shared-bike operators need to determine optimal truck routes for bike repositioning (BR). The availability of bikes at different bike stations and the cost of repositioning operations are the primary concerns. Typically, in urban cities, the size of the BSN is large. As the available resources are limited (e.g., bikes, repositioning trucks and truck capacity, budget for operating the BSS), it is challenging for the operators to reposition bikes efficiently (Haider et al., 2018; Raviv and Kolka, 2013; Schuijbroek et al., 2017). For instance, in Paris, repositioning trucks travel long distances between large bike stations near train stations or universities (Legros, 2019). The repositioning operation exhibited a low efficiency, but imposed a high cost. In this study, we aim to decompose the large BSN into smaller sub-networks for BR. A clustering approach based on community detection is proposed to achieve the goal, thereby facilitating the logistics and operations management related to BR in BSSs.

Different from other transportation and logistics networks, in our research, the connectivity between the bike stations is defined by not only the road infrastructure (i.e., roads and bike lanes), but also bike trips. Instead of considering distances between bike stations and the number of bikes available at each station as in

the majority of existing studies, our research utilizes bike flows between stations. Furthermore, the objective of most of the existing mathematical models is to minimize the storage cost or total distances between bike stations in clusters, but has not considered the characteristics of BSSs as a transportation mode. The existing methods are mostly applied to smaller-scale instances of BSSs with the number of bike stations (NBS) ranging from tens to hundreds. In real-life large-scale BSSs (e.g., NBS $\geq$ 1000), decomposition of the network is required for practical deployment of the solution method.

Thus, there are several open questions for this research: (1) How should bike stations be clustered for effective BR? (2) What are the optimal clusters for BR? (3) Do clusters suggested by user demands help the analysis of travel patterns in a large-scale BSS? To address these three questions, this paper aims to establish a new demand-driven clustering framework for BSNs and BR. The objectives are listed below:

- To utilize user trip information for effective BR in large BSNs;
- To develop a mathematical model and the required solution method to determine optimal clusters of bike stations for effective BR; and
- To demonstrate the effectiveness and efficiency of our solution method by conducting a case study of the world's largest BSN.

Referring to the objectives of this research, we develop a demand-driven approach to cluster bike stations in BSNs for understanding spatiotemporal patterns of user demands, thereby enhancing the efficiency of BR operations. First, a directed graph is constructed to represent the BSN, whose vertices are bike stations and arcs represent bike flows, weighted by the number of trips between the bike stations. Second, we propose a mixed integer programming (MIP) model to minimize the difference between the inflow and outflow of bikes (DIOB) for each cluster in the BSN. A novel demand-driven algorithm based on community detection is developed to solve the problem. A computational study is conducted to assess the effectiveness and computational efficiency of our solution approach. Third, we carry out an analysis with data captured from the world's largest BSS, which consists of more than 3,000 bike stations. The clusters identified from bike trips during the morning peak and the evening peak are compared and the characteristics of clusters (e.g., the distances between bike stations in the same cluster and the numbers of trips staying within and traveling out of each cluster) are presented.

This paper has three main contributions that distinguish our research on the operations of BSNs and BR from the existing studies, which are to be presented as follows.

(i). This research focuses on a different problem for BR. Contrary to most of the existing studies on BR, which focus on bike loading and unloading problems formulated as variants of VRP, we study the bike station clustering problem. We propose a demand-driven clustering framework for BSNs, considering the directions and amounts of user demands. The clusters detected by our approach exhibit a combination of essential characteristics of BR, the balance of bike flows and geographical proximity. These characteristics are mostly considered as the key objectives of BR (Szeto and Shui, 2018).

(ii). We introduce a definition of bike station cluster and a novel mathematical formulation, which captures user bike flows from BSS operational data, for BR planning. The objective of our model is different from traditional ones, which consider traveling distance or operational costs, but incoporates the user trip information for clustering. A cluster with small DIOB can be considered as a relatively independent sub-network. The operator can, therefore, perform BR within the cluster without considering BR activities in other clusters. Thus, this approach is expected to be effective in preventing oversupply of bikes and long-distance repositioning. Consequently, the service level of a BSS, which is an important measure of the success of a BSS (Kabra et al., 2019), can be improved.

(iii). We demonstrate the effectiveness and efficiency of our proposed methodology with a case study of the world's largest BSS. The results show the performance of the proposed approach and the effectiveness of clustering by bike trip records as an alternative to determining clusters of bike stations. Taking into account the interdependencies among user demands at service points, this study illustrates the travel patterns of bike users by detecting representative clusters in different time periods.

Table 1 presents the abbreviations used in this paper.

The rest of this paper is organized as follows. Section 2 presents a review of the existing literature on clustering for BSNs and BR. Section 3 presents the problem description and model formulation. Section 4 describes the proposed community detection approach to clustering the bike stations. In Section 5, we report on the computational efficiency of the proposed procedure. A case study of the Hangzhou public BSS is conducted to illustrate the performance of our solution framework. The paper is concluded in Section 6 with a summary of the research and a discussion about the modeling issues and several managerial implications.

Table 1: Abbreviations used in this paper.

BSS	Bike Sharing System
BSN	Bike Station Network
BR	Bike Repositioning
BRP	Bike Repositioning Problem
DDC	Demand-driven Clustering
DIOB	Difference between Inflow and Outflow
FUA	Fast Unfolding Algorithm
LPA	Label Propagation Algorithm
MCESS	Minimum Cut into Equal-Sized Subsets
MFUA	Modified Fast Unfolding Algorithm
NBS	Number of Bike Stations
VRP	Vehicle Routing Problem

2 Literature Review

To distinguish our work from previous literature, the existing research with the application of clustering techniques for BR in BSSs and the solution methodologies for managing operations in large-scale BSSs is reviewed.

Clustering techniques applied for BSSs mainly focus on the analysis of travel patterns, particularly the relationships between bike trips and various exogenous variables. Froehlich et al. (2009) analyze the Barcelona's BSS by dendrogram clustering and uncover the daily demand patterns and behavioral patterns in different areas. Zhu and Diao (2019) group bike stations based on daily temporal travel patterns by fuzzy clustering. Gervini and Khanal (2019) apply hierarchical clustering to explore the user demand pattern based on the characteristics of the bike trips. Du et al. (2019) investigate the bike trip patterns around subway stations by hierarchical clustering. Their proposed clustering-based time-domain analysis method is tested with the data of Mobike shared bikes in Shanghai. Existing studies apply clustering techniques for a better understanding of the bike usages with geographic proximity, demographic characteristics, users' characteristics, and weather conditions; however, demand clustering has not been formally applied to BR in the literature. In our research, demand clustering by community detection is successfully applied to a large-scale real-world network for more efficient management of the BSS.

Previous studies on clustering for BR are still inadequate. Forma et al. (2015) develop a 3-step heuristic for the BR problem. An MIP model for clustering bike stations is proposed in the first step, considering the bike inventory level and geographic information. The repositioning routing, which allows vehicles to travel between clusters, is determined in the second and third steps. Their clusters are constructed to decompose

the large network into smaller sub-networks. Schuijbroek et al. (2017) propose a cluster-first route-second heuristic to determine the service level requirements at each bike station and the vehicle routes. Their clustering problem simultaneously considers the service level requirement and approximate routing costs by a new maximum spanning star approximation. However, the user demands in BSSs have not been well considered for clustering bike stations for BR. To our best knowledge, Lahoorpoor et al. (2019) is the first and only existing work on bike station clustering for BR, where the origin-destination (OD) information is utilized. They present a similarity measure method based on the trips between stations to increase high intra-cluster trips, using a hierarchical agglomerative clustering method. A case study is conducted on a real-world BSS, which consists of 582 bike stations. From their results, the numbers of bike stations in each cluster range from 14 to 58. Existing research shows that solving BR problems by clustering bike stations in BSN could successfully enhance the scalability of the solution method.

The bike repositioning problem (BRP) has attracted most of the researchers' interest in recent years. Compared to traditional logistic problems, there are three distinctive characteristics of BRPs. First, bikes in BRPs are different from goods in traditional logistics problems, as bikes are ridden by cyclists as means of transportation. Bikes are mobilized by both users and the operator, while goods are delivered only by the operators. Second, the imbalance between bike supply and user demand results from the availability of bikes, numbers of docks at bike stations, and bike flows. Third, bike stations are both demand points and supply points in BSNs, while depots are always different from service points in typical logistics networks. Thus, the availability of bikes can influence the users' travel behavior and also the demands for bikes and docks. All of these characteristics of BSSs motivate us to utilize bike trip information to determine better clusters to optimize decisions in BRPs.

Most of the existing studies consider BRPs as vehicle routing problems (VRPs) or traveling salesman problems (TSPs). Similarly, the objectives of BRPs focus on the efficiency and impacts of BR strategies. The objectives include minimization of unmet user demand or overall user dissatisfaction (Alvarez-Valdes et al., 2016; Forma et al., 2015; Haider et al., 2018; Legros, 2019), minimization of travel cost or total cost (Chemla et al., 2013; Dell'Amico et al., 2018; Erdoğan et al., 2015; Zhang et al., 2017), minimization of penalty costs, traveling time, or service time (Ho and Szeto, 2014; Li et al., 2016; Raviv et al., 2013; Du et al., 2020), and minimization of the maximum tour length or total traveling distance (Forma et al., 2015; Schuijbroek et al., 2017). Most studies on BRPs adopt the above objectives, as the efficiency or accessibility of a BSS is a key performance measure.

While the literature on BRP and demand analysis for BSSs is extensive (Albiński et al., 2018), only a few studies focus on the analysis and operations management of large-scale BSSs. Shaheen et al. (2011) conduct a survey to investigate the usage pattern of the Hangzhou BSS. They find that people use the BSS after riding other transportation modes (e.g., bus, car, and taxi). The most frequently-used bike stations are closest to either home (40%) or work (40%). Kabra et al. (2019) analyze how the accessibility and availability of bikes influence the performance of a BSS. The estimates are obtained using data from the Vélib' system in Paris, which is the largest BSS out of China. Almost 80% of rides travelled less than 300 m per trip. 10% more bikes at bike stations could induce demands by 12.211%. Their study highlights that large-scale BSSs need to improve the service level. Yang et al. (2019) develop a data-driven simulation-based approach for predicting users' demand in BSSs and evaluate its effectiveness with data also from the Hangzhou BSS. In these studies, the focuses are mainly on understanding the user demands and travel behaviors; however, the analyses have not been further utilized for clustering bike stations.

Community[1] structure theory has been applied in transportation systems. Mesa-Arango and Ukkusuri (2015) explore the problem of clustering demand in freight logistics networks by community detection. Du et al. (2018) consider delay propagation among airports and apply a community detection algorithm to decompose a large aviation network into several sub-regions. Van Nguyen et al. (2020) develop a two-stage approach by integrating data mining and community structure theory to determine optimal locations and service areas of dry ports in a large-scale inland transportation system. For BSSs, Zhou (2015) applies community detection on a massive dataset of the BSN in Chicago to illustrate the bike flow dynamics on weekdays and weekends as well as different travel patterns by subscribers and other users. Zhang and Meng (2019) develop a bike allocation strategy in a competitive free-floating bike sharing market based on community structure theory. However, the existing research has not formally considered characteristics of bike flows to detect communities and apply such approach , for the purpose of BR.

In summary, the majority of the research on BR focuses on the daily repositioning operations which are variants of VRPs or TSPs. There are a few studies which apply clustering techniques for BR, there lacks a more formal definition of optimal clusters and bike user behaviors are rarely considered. Our research distinguishes from the existing studies as follows. First, we propose a novel optimization model, powered by large volumes of user trip data, for optimal clustering for BR operations. Second, a modified fast unfolding algorithm (MFUA) is proposed to tackle large-scale real-world instances. Third, our computational

[1]The community termed in this paper is also referred as the cluster.

experiments demonstrate that the optimal clusters identified by our proposed solution method preserve the geographic proximity between bike stations, which is an essential element in BR.

3 Problem Definition and Model Formulation

In practical BR operations, when the coverage of bike stations is large and the user demands are high, the operator of the BSS may need to deploy multiple trucks for BR in different regions. Thus, BSS operators typically consider their BR operations a two-phase problem. As Figure 1 illustrates, the first step is to cluster the bike stations such that each cluster can be served by a truck and the second step is to deploy trucks for repositioning of bikes (i.e., loading and unloading operations). In this paper, we focus on the first problem – bike station clustering – and will refer the reader to the studies on the second problem of bike loading & unloading, which have been extensively studied as variants of VRPs and solved by various solution methods (e.g., Ho and Szeto, 2014; Li et al., 2016; Dell'Amico et al., 2018; Raviv et al., 2013; Du et al., 2020).

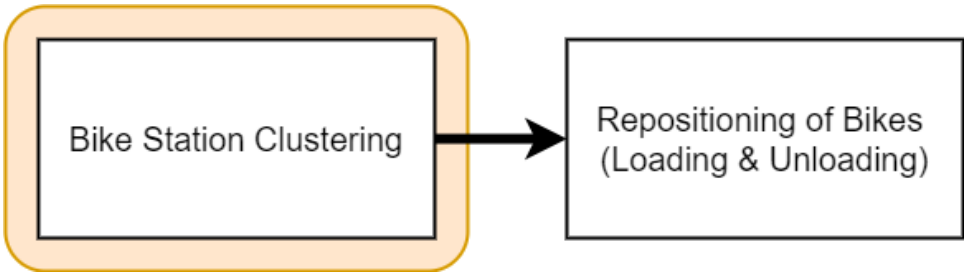


Figure 1: A two-phase approach to bike repositioning operations.

This paper proposes a demand-driven approach for clustering bike stations in a large-scale BSN for improving the efficiency of BR. A mathematical model to minimize the DIOB by user demands is developed based on the proposed directed network. Due to dynamic demands in various locations, the operator needs to rebalance bike supply and user demands. The different clusters of bike stations, optimal routes within each cluster, and also the loading/unloading activities for bike repositioning should be determined based on the whole BSN. Particularly for dynamic bike repositioning, trucks are supposed to serve for a small number of bike stations. Long travel distances for BR shall be avoided. On the other hand, the user demands shall also be a critical factor in BR operations, as such demands influence the number of bikes that can be loaded to the truck at each station. All of the above are important factors that determine the effectiveness of a BR plan.

A key challenge faced by the operator is how to determine optimal bike station clusters for BR. Our

research proposes a clustering method driven by the estimated bike flows in the time period. The algorithm determines the clusters by minimizing the difference between the inflow and outflow of bikes in each cluster. The operator can, therefore, allocate repositioning trucks to these clusters. By doing so, the availability of bikes in each cluster is expected to be more stable. Second, the travel distance of a bike trip is typically short (Hasija et al., 2020). Therefore, clustering by the volumes of user trips between bike stations can ensure both the matching of supply and demand and also the geographic proximity between stations within the same cluster.

Several constraints should be considered for clustering bike stations. For practical task assignments, each repositioning truck serves bike stations of exactly one cluster (Dell'Amico et al., 2014); that is, a truck will not visit any bike stations in other clusters (see Figure 2). It is common in practice and helps to avoid repeated services. The number of repositioning trucks, the required time for bike loading and unloading at each station, and the service radius of the repositioning trucks impose upper bounds on the number of bike stations that can be assigned to a cluster.

In this work, we consider a directed network $G = (S,\ A,\ W)$, $W : A \rightarrow \mathbb{R}_+ \cup 0$, whose vertices are bike stations and arcs represent bike flows. In our proposed framework, we refer demands to the bike trips *actually* realized as these trips change the locations of the bikes such that repositioning is required. There could be unsatisifed demands due to unavilability of bikes; however, these demands would not change the locations of the bikes such that repositoning operations are not affected. The set of vertices is denoted by S, indexed by $s, r \in S$, and the set of arcs is denoted by A, indexed by (s, r). Arc (s, r) indicates that the user bike flow from station s to station r exists and is weighted by the number of trips in a time period along the direction $w_{sr} = W(s, r)$. In our case study to be presented in Section 5, the time period is set to the hour right before the time of making bike station clustering decisions. The rationale is that we aim to maintain a relatively stable total number of bikes within a cluster such that the supply of bikes to the stations with shortages can be self-sustained; repositioning trucks are not required to travel to another cluster to pick up bikes to satisfy the demands in its own service region. This can be achieved by minimizing DIOB resulting from the previous period. The choice of the length of a period shall be long enough to accommodate sufficient bike trips for estimating the bike flows but, at the same time, not too long for capturing the time-varying bike demands and flows. In many bike systems, the length of a pricing interval is usually set to 15 minutes (e.g., Metropolradruhr in Germany (Metropolradruhr, 2021)), 30 minutes (e.g., Meituan Bike in Mainland China (Meituan Bike, 2021); Locobike in Hong Kong SAR (Locobike, 2021); and Santander

Cycle in London (Santander Cycles, 2021)), or 45 minutes (e.g., Citi Bike in New York City (Citi Bike, 2021); Bay Wheels in San Francisco (Bay Wheels, 2021)). The length of the pricing interval can reflect the typical ride time of a user in the region. The time-varying demand patterns are typically reflected on an hourly basis. Thus, in our case study, we set the length to be one hour. Nevertheless, our solution methodology is general for other choices of time interval lengths.

We denote n^{min} and n^{max} the minimum and maximum numbers of bike stations that can be assigned to a cluster. n^{min} can be interpreted as the minimum number of bike stations that a repositioning truck has to service for a reasonable utilization, due to the limited number of trucks. n^{max} can be interpreted as a limit such that repositioning trucks are not overloaded and the repositioning operations are not delayed. Our model is generic that, even if there are no bounds on the number of bike stations in each cluster, n^{min} and n^{max} can be set to 0 and $|S|$, respectively. The set of clusters is denoted by C. Note that $\cup_{c \in C} c = S$ and $c_1 \cap c_2 = \emptyset \; \forall c_1, c_2 \in C, c_1 \neq c_2$.

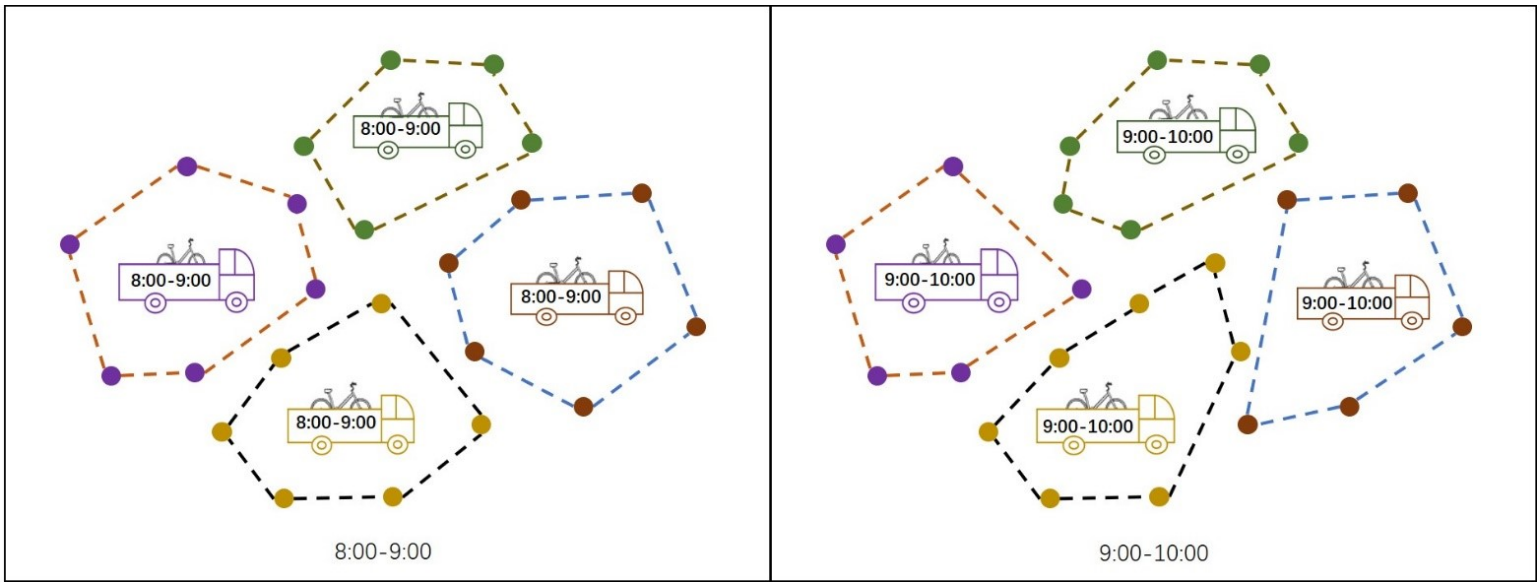


Figure 2: Dynamic bike station clustering for bike repositioning.

Our formulation includes the following **decision variables:**

$$x_s^c = \begin{cases} 1 & \text{if vertex } s \text{ is assigned to cluster } c, \\ 0 & \text{otherwise;} \end{cases}$$

$$y_c = \begin{cases} 1 & \text{if cluster } c \text{ is formed,} \\ 0 & \text{otherwise.} \end{cases}$$

Table 2: Notations used in this paper.

Sets & Parameters	
$G = (S,\ A,\ W)$	A directed graph which represents the bike station network
S	Set of vertices (bike stations)
A	Set of arcs (connections between bike stations)
C	Set of clusters
(s, r)	Arc from node s to node r
$W(s, r)$ or w_{sr}	Amount of bike flow from node s to node r
n^{min}	Minimum number of bike stations in a cluster
n^{max}	Maximum number of bike stations in a cluster
Decision Variables	
x_s^c	1 if vertex s is assigned to cluster c; 0 otherwise.
y_c	1 if cluster c is formed; 0 otherwise.
Q	Absolute difference between inflow and outflow of bikes to and from the cluster, summing over all clusters
Q_c	Absolute difference between inflow and outflow of bikes to and from cluster c

Based on the problem characteristics, the demand-driven bike station clustering problem can be formulated as follows:

$$\min Q = \sum_{c \in C} | \sum_{s,r \in S} w_{sr} \left(x_r^c - x_s^c\right)| \tag{1}$$

subject to

$$\sum_{c \in C} x_s^c = 1 \qquad \forall s \in S \tag{2}$$

$$\sum_{s \in S} x_s^c \leq M \cdot y_c \qquad \forall c \in C \tag{3}$$

$$\sum_{s \in S} x_s^c \leq n^{max} \qquad \forall c \in C \tag{4}$$

$$n^{min} \cdot y_c \leq \sum_{s \in S} x_s^c \qquad \forall c \in C \tag{5}$$

$$x_s^c \in \{0, 1\} \qquad \forall s \in S, c \in C \tag{6}$$

$$y_c \in \{0, 1\} \qquad \forall c \in C \tag{7}$$

where M is a sufficiently large number.

Objective (1) is to minimize the absolute difference between inflow and outflow of bikes to and from the cluster, summing over all clusters. In Objective (1), the term

$$w_{sr}\left(x_r^c - x_s^c\right) = \begin{cases} w_{sr} & \text{if vertex } r \text{ is in cluster } c \text{ and vertex } s \text{ is not,} \\ -w_{sr} & \text{if vertex } s \text{ is in cluster } c \text{ and vertex } r \text{ is not,} \\ 0 & \text{if both are in cluster } c \text{ or both are not in cluster } c. \end{cases}$$

Thus, the term $|\sum_{s,r\in S} w_{sr}\left(x_r^c - x_s^c\right)|$ measures the sum of inflows and outflows of bikes to and from Cluster c, i.e., DIOB. Constraints (2) state that each bike station is allocated to only one cluster. Constraints (3) state that a cluster is formed only when there is more than one station in a cluster. Constraints (4) and (5) require that the number of bike stations in each cluster should be within the bounds. Constraints (6) and (7) define the domain of variables x_s^c and y_c. To summarize, the model aims to allocate stations to clusters such that the difference between inflow and outflow of bikes to and from the cluster is minimized, where the minimum and maximum numbers of bike stations in a cluster are respected.

The above mathematical model is a non-linear MIP due to the non-linearity of Objective Function (1). Linearization techniques are applied to the model to obtain the following linear MIP:

$$\min Q = \sum_{c\in C} z_c \tag{8}$$

subject to

$$\sum_{c\in C} x_s^c = 1 \qquad \forall s \in S \tag{9}$$

$$\sum_{s,r\in S} w_{sr}\cdot\left(x_r^c - x_s^c\right) \le z_c \qquad \forall c \in C \tag{10}$$

$$-\sum_{s,r\in S} w_{sr}\cdot\left(x_r^c - x_s^c\right) \le z_c \qquad \forall c \in C \tag{11}$$

$$\sum_{s\in S} x_s^c \le n^{max} \qquad \forall c \in C \tag{12}$$

$$\sum_{s\in S} x_s^c \le M\cdot y_c \qquad \forall c \in C \tag{13}$$

$$n^{min}\cdot y_c \le \sum_{s\in S} x_s^c \qquad \forall c \in C \tag{14}$$

$$x_s^c \in \{0,1\} \qquad \forall s \in S, c \in C \tag{15}$$

$$y_c \in \{0,1\} \qquad \forall c \in C \tag{16}$$

$$z_c \ge 0 \qquad \forall c \in C \tag{17}$$

This bike station clustering problem is computationally challenging, as shown in Proposition 1. Thus,

a computationally efficient solution methodology is proposed in Section 4.

Proposition 1. *The demand-driven bike station clustering problem is NP-hard.*

Proof: We consider a polynomial time reduction from a NP-hard problem, the Minimum Cut into Equal-Sized Subsets problem (MCESS) (Garey et al., 1976): Given a graph $G(S, A)$ and positive integer u, is there a partition c_1 and c_2, which satisfies $c_1 \cup c_2 = S$ with $c_1 \cap c_2 = \emptyset$, $|c_1| = |c_2|$, and $|(s, r) \in A : s \in c_1, r \in c_2| \leq u$.

Given an instance of the MCESS problem, we define an instance of (DDC) with a directed network $G = (S,\ A,\ W)$. If $|S|$ is an odd number, there is no feasible solution. If $|S|$ is an even number, we set $w_{ij} = 1, \forall w_{ij} \in W$ and $n_{min} = n_{max} = |S|/2$. Such settings reduce the MCESS problem (Garey et al., 1976) to the DDC problem. Then, if the optimal value of the DDC problem is not greater than u, the answer of MCESS problem is 'YES' and vice versa. The MCESS problem is now interpreted as a DDC problem. Thus, the DDC problem is NP-hard. □

4 Solution Methodology

In this section, we present our proposed modified fast unfolding algorithm (MFUA) for solving the clustering problem presented in Section 3.

Community structure in the real-world network was defined by Girvan and Newman (2002). It is one common characteristic of complex networks (Zhang and Meng, 2019). It reveals how complex networks are composed of relatively independent and interlaced sub-networks. The connections between the vertices in each community are close, while the connections between communities are relatively sparse. The community structure makes the generation and evolution of system much quicker and more stable than if the system is unstructured (Zhou et al., 2012; Mesa-Arango and Ukkusuri, 2015). Considering a BSN as a complex network with bike stations and bike flows being vertices and arcs, respectively, the BSN also exhibits the community structure. As a short-distance transportation mode, shared bikes are typically distributed within a fixed area since users do not use bikes for long-distance travels. While bike stations are connected by road infrastructure, their connectivity in our graph is defined by user demands.

To deliver practical solutions, we propose MFUA for tackling the community detection problem. Blondel et al. (2008) propose a fast unfolding algorithm (FUA), which is developed based on modularity optimization, to uncover the community structure in an undirected graph and examine the effectiveness of their approach in several large-scale networks. Fortunato (2010) discusses the principles and strengths of FUA

and its good performance on large networks. Compared to the other known community detection methods (e.g., Newman (2004)), FUA can be applied in community detection of weighted undirected networks more efficiently. For instance, compared with the Label Propagation Algorithm (LPA) (Rahjavan et al., 2007), FUA is more appropriate to be applied to BSNs than LPA. The main idea of LPA is based on the connections between nodes. If there is an existing arc, the label of one node would be the same as its neighboring node. The computational complexity of LPA is lower than that of FUA. However, LPA depends on only the network structure, being less effective than to be applied to directed and weighted networks.

The original FUA proposed by Blondel et al. (2008) aims to determine high modularity partitions of large undirected networks. FUA is an iterative heuristic which implements two procedures at each iteration: (i) (re-)allocation of nodes to clusters and (ii) aggregation of nodes to form new nodes. FUA identifies final clusters of nodes at its convergence (i.e., no changes in cluster formation from one iteration to the next). For (re-)allocation of nodes to cluster, modularity, which is a metric of the strength of the partition of a network, is used to determine if the (re-)allocation of one node from its original cluster to another is beneficial. In their proposed algorithm, the change of modularity is computed by moving one node from its original cluster to another one. In other words, such a change would only depend on the reallocation of an individual node to other clusters for simplified computational complexity. Once changes of modularity are computed, new clusters are formed based on the maximized modularity and nodes within each cluster will be aggregated as a new node. Our proposed algorithm shares the key ideas of FUA – (i) (re-)allocation of nodes to clusters and (ii) aggregation of nodes to form new nodes – which help unfold a complete hierarchical community structure for the network. Nevertheless, in our application, the bike flows are directed and our problem objective is to minimize the difference between inflow and outflow of bike trips in a cluster. To this end, we modify the FUA and develop MFUA to generalize the approach for weighted directed graphs. In particular, FUA has been generalized for directed networks (Chang et al., 2017). In our application for bike station clustering, we calculate the change of DIOB by (re-)allocating a node from its original cluster to a new one to assess if this new formation of clusters is beneficial.

We first define the absolute difference between the outflow and inflow of bikes of a given cluster $c \subseteq S$, denoted by

$$Q_c = | \sum_{s \in c, r \notin c} w_{sr} - \sum_{s \in c, r \notin c} w_{rs} | \tag{18}$$

Proposition 2. *Q_c can be computed in a more computationally efficient way as follows.*

$$Q_c = \mid \sum_{s\in c}(\sum_{r\in S} w_{rs} - \sum_{r\in S} w_{sr})\mid \tag{19}$$

Proof: We assume that there are m bike stations ($s_i,\ i = 1, 2, 3, \ldots, m$) in cluster c. The inflow of bikes from cluster c is calculated by

$$\sum_{s_i\in c, r\in S\setminus c} w_{rs_i} = \sum_{r\in S\setminus c} w_{rs_1} + \sum_{r\in S\setminus c} w_{rs_2} + \sum_{r\in S\setminus c} w_{rs_3} + \ldots + \sum_{r\in S\setminus c} w_{rs_m} \tag{20}$$

The outflow of bikes to cluster c is calculated by

$$\sum_{s_i\in c, r\in S\setminus c} w_{sr} = \sum_{r\in S\setminus c} w_{s_1r} + \sum_{r\in S\setminus c} w_{s_2r} + \sum_{r\in S\setminus c} w_{s_3r} + \ldots + \sum_{r\in S\setminus c} w_{s_mr} \tag{21}$$

Q_c is calculated by

$$Q_c = \mid \sum_{r\in S\setminus c} w_{rs_1} - \sum_{r\in S\setminus c} w_{s_1r} + \ldots + \sum_{r\in S\setminus c} w_{rs_m} - \sum_{r\in S\setminus c} w_{s_mr}\mid \tag{22}$$

Note that due to the conservation of bike flows within cluster c, we have:

$$\sum_{s_i\in c} w_{s_is_1} - \sum_{s_i\in c} w_{s_1s_i} + \sum_{s_i\in c} w_{s_is_2} - \sum_{s_i\in c} w_{s_2s_i} + \ldots + \sum_{s_i\in c} w_{s_is_m} - \sum_{s_i\in c} w_{s_ms_i} = 0 \tag{23}$$

Substituting Equation (22) into Equation (23), we have:

$$\begin{aligned} Q_c = &\mid (\sum_{r\in S\setminus c} w_{rs_1} - \sum_{r\in S\setminus c} w_{s_1r} + \sum_{s_i\in c} w_{s_is_1} - \sum_{s_i\in c} w_{s_1s_i}) + \ldots \\ &+ (\sum_{r\in S\setminus c} w_{rs_m} - \sum_{r\in S\setminus c} w_{s_mr} + \sum_{s_i\in c} w_{s_is_m} - \sum_{s_i\in c} w_{s_ms_i})\mid \\ = &\mid \sum_{s\in c}(\sum_{r\in S} w_{rs} - \sum_{r\in S} w_{sr})\mid \end{aligned} \tag{24}$$

□

Proposition 2 shows that the difference between the inflow and outflow of bike trips of a cluster is simply the absolute value of the sum of the net bike flows over all the bike stations in that cluster. Thus, when evaluating Q_c, one may not need to consider all the cuts resulting from c and $S \setminus c$, but sum the net

bike flows over all the bike stations in c. This property helps reduce the computational efforts of MFUA.

We further define $\Delta Q^s_{c,k}$ $(c, k \subseteq S, s \in c, s \notin k)$ as

$$\Delta Q^s_{c,k} = Q_{k\cup\{s\}} - Q_k + Q_{c\setminus\{s\}} - Q_c \tag{25}$$

$\Delta Q^s_{c,k}$ can be interpreted as the change in the value of Q when moving vertex s from cluster c to cluster k.

The main idea of MFUA is as follows. Suppose that we start with a directed weighted graph $G(S, A, W)$. Initially, each cluster is set to a different vertex. Then, the algorithm iterates according to the following procedure. For each vertex s, the algorithm considers moving it to the community of each vertex r connected to s, while ensuring Constraints (2) are satisfied. Vertex s will be grouped to the neighboring community, which achieves the best improvement in the objective value. If no improvement is made when merging vertex s with any neighboring community, it remains in the current one. If a community becomes an empty set after the removal of a vertex, it will be deleted from the set of communities. This process is repeated sequentially for all vertices until no further improvement can be achieved. To this end, we consider a new graph, whose vertices are now the communities formed and arcs represent the bike flows between the communities, weighted by the number of bike trips between each pair of communities. After this new graph is constructed, the same sequence of steps repeats until the constraints are satisfied.

Figure 3 provides an example of the implementation of MFUA on a small BSN for illustration purposes. At each iteration, each node is sequentially assigned to a community according to the best improvement in the objective function, and a new community is formed by aggregating the nodes assigned to it. After the second iteration, no improvement in the objective value can be made by further allocation of nodes to communities, and therefore, MFUA terminates and the final communities of bike stations are formed. The detailed flow of MUFA is depicted in Algorithm 1.

5 Case Study

5.1 Case Description

We conduct a case study with computational experiments based on a dataset collected from the Hangzhou (China) public BSS. It is the world's largest public BSS and officially began operations on September 16, 2008. It is a SBBSS, which allows users to rent and return bikes only at the stations. As of December 2018, there were nearly 1.2 million users, 4,198 bikes stations, and nearly 101,700 public bikes registered. The

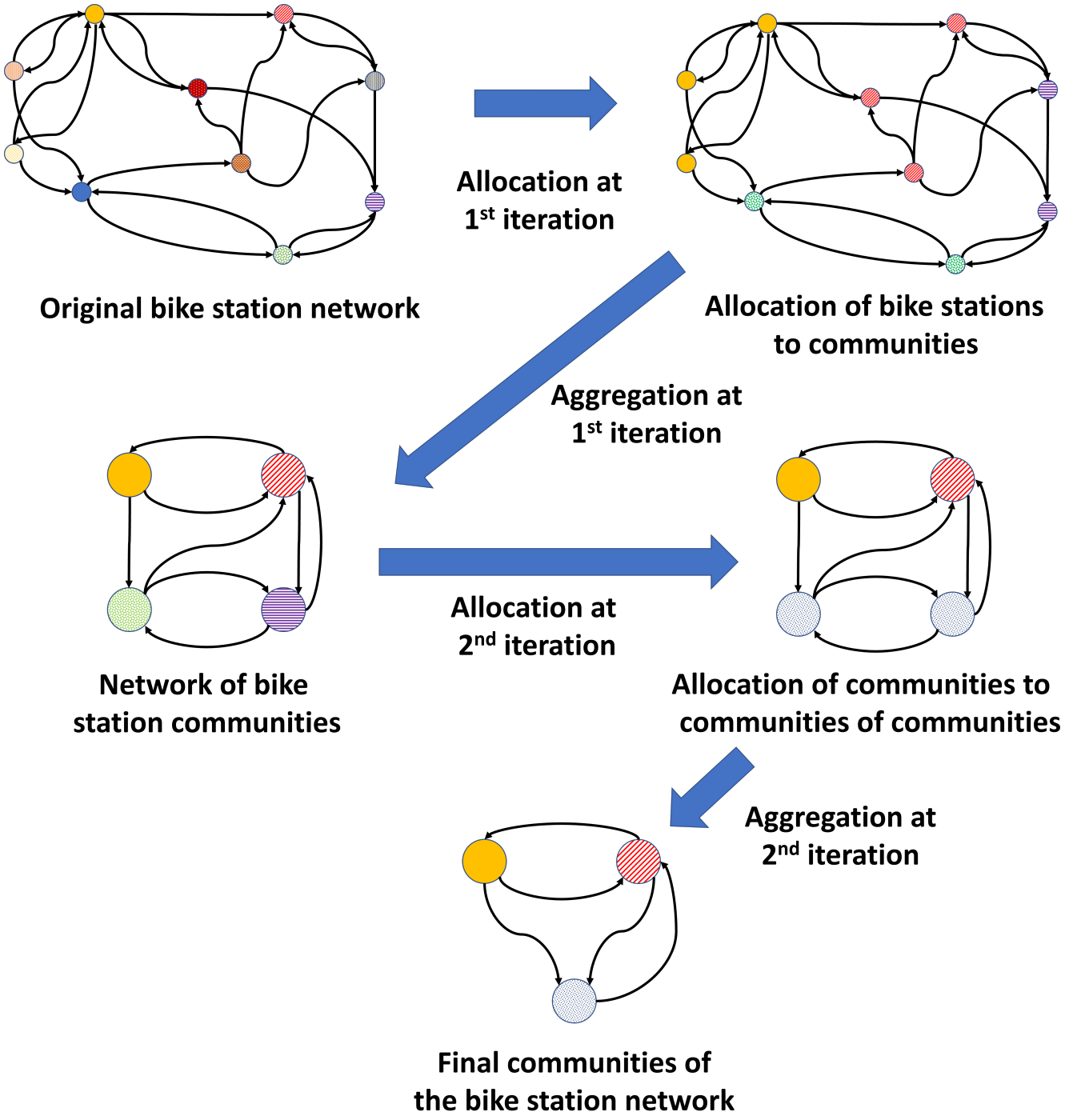


Figure 3: An illustration of our proposed modified fast unfolding algorithm (MFUA). Nodes of the same color/shade are in the same community. An arc between nodes indicates that there is bike flow from one node to another (in the same direction). In this example, for illustration purposes, only two iterations are required to obtain the clusters of bike stations.

Algorithm 1 Modified Fast Unfolding Algorithm (MFUA)

Require: $G(S, A, W)$
Initialize C
$C = S$
$X \leftarrow |S| \times |S|$ **matrix.**
for each $i \in S,\ c \in C$ **do**
  if $i = c$ **then**
    $x_i^c = 1$
  else
    $x_i^c = 0$
  end if
end for
Initialize $\gamma < 0, c^{'}, c^{''}, c^{'''}$
while $\gamma < 0$) **do**
  $\beta = \sum_{c \in C} Q_c$
  for $i \in S$ **do**
    $\Delta Q^i = 0$
    for $j \in S$ **do**
      if $w_{ij} + w_{ji} > 0$ **then**
        for $c \in C$ **do**
          if $x_i^c = 1$ **then**
            $c^{'} = c$
          end if
          if $x_j^c = 1$ **then**
            $c^{''} = c$
          end if
        end for
        Compute $\Delta Q^i_{c^{'},c^{''}}$
        if $\Delta Q^i_{c^{'},c^{''}} < \Delta Q^i$ and $n^{min} \leq \sum_{i \in S} x_i^{c^{'}} + \sum_{i \in S} x_i^{c^{''}} \leq n^{max}$ **then**
          $\Delta Q^i = \Delta Q^i_{c^{'},c^{''}}$ and $c^{'''} = c^{''}$
        else
          $c^{'''} = c^{'}$
        end if
        $x_i^{c^{'}} = 0$ and $x_i^{c^{'''}} = 1$
      end if
    end for
  end for
  for $c \in C, i \in S$ **do**
    if $\sum_{i \in S} x_i^c = 0$ **then**
      $C = C \setminus \{c\}$
    end if
  end for
  $W \leftarrow |C| \times |C|$ **matrix.**
  for $c \in C, k \in C$ **do**
    $w_{ck} = \sum_{c,k \in C} \sum_{i,j \in S} w_{ij} \cdot x_i^c \cdot x_j^k$
  end for
  $S \leftarrow C$
  $\alpha = \sum_{c \in C} Q_c$
  $\gamma = \alpha - \beta$
end while
Ensure: X

maximum number of trips made in a day reached 473,000 person-times with free usage rate exceeding 96%.

The data about the bike trips in the Hangzhou public BSS were collected from smart IC card records upon bike retrieval and return. Each transaction contains the following information: operation ID (a pick-up or drop-off event), bike ID, user ID or operator ID, station ID, timestamp, and fare.

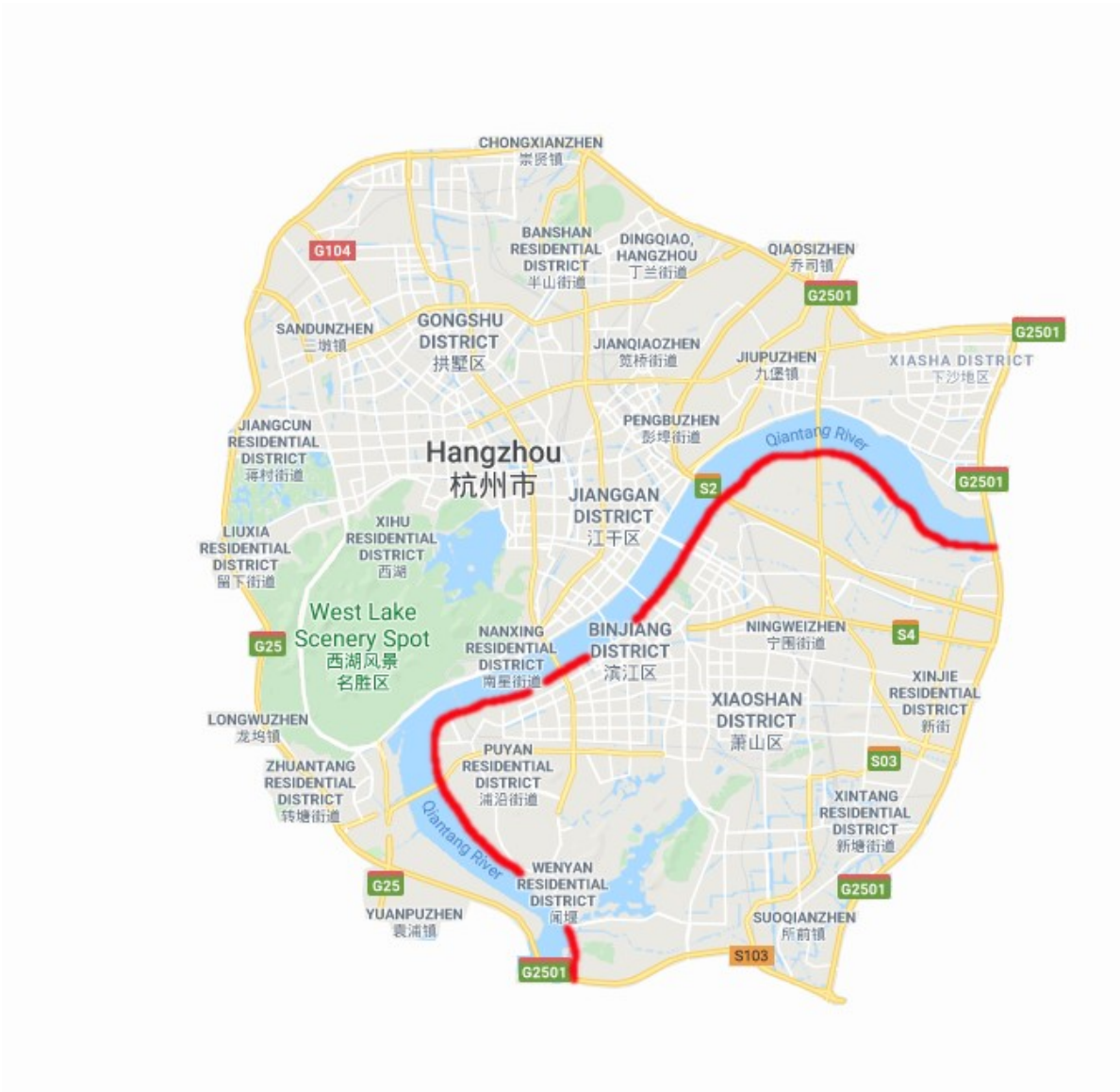


Figure 4: Hangzhou City Map (Map Data: Google Maps 2020).

5.2 Experiment Design

In this study, we consider that each bike trip is characterized by origin, destination, departure time, and arrival time. We grouped the transaction records on May 15th (Monday), 2013, by the departure hour for the

case study. There were 2,712 bike stations, in total, which had bike retrieval/return records. In each hour, bike stations with no transaction record were excluded from our computational instances. After processing the data, the numbers of bike stations included in our experiments in an hour vary from 1,896 to 2,471. The experiments are conducted using datasets of two morning peak hours (07:00-08:00, 08:00-09:00), two off-peak hours (10:00-11:00, 14:00-15:00), and two evening peak hours (17:00-18:00, 18:00-19:00).

In the case study, we conduct computational experiments to examine (i) computational effectiveness and efficiency of our proposed methodology, (ii) effects of the bounds on cluster size, and (iii) the characteristics of the clusters in different periods.

Finally, we analyze the characteristics of the clusters in different hours found by our method and discuss the benefits of clusters used for BR. Also, clusters formed during different periods are compared, and the similarity of clusters can suggest the user ride behaviors during the period.

5.3 Computational Effectiveness and Efficiency

We first compare the solutions resulting from MFUA with the ones from CPLEX. Optimality of CPLEX was guaranteed as a branch-and-cut approach was applied to solve the problem. And then we compare the results obtained by MFUA and the K-means algorithm, a popular clustering method for transportation and logistics applications. MFUA (coded in Java), CPLEX, and K-means clustering were run on a Dell OptiPlex 7040 desktop with an Intel Core i7-6700 CPU@ 3.4 GHz and 16.0 GB RAM.

Computational instances of this set of experiments were constructed based on smaller networks associated with bike stations in southeast area of Qiantang River (bounded by the red line in Figure 4). From the dataset, we observe that there was only a small number of bike trips traveling from one side of Qiantang River to another. In the southeast of Hangzhou bounded by Qiantang River, there were less than 200 bike stations. We construct the computational instances of different sizes by selecting bike stations from this region. Bike trip data from 07:00-08:00 and 17:00-18:00 are used to construct the weights for the morning and evening peak hours, respectively.

The performance indicators of interest are the average absolute DIOB per cluster, denoted by $Avg.Q_c$, obtained by the two approaches, and the CPU seconds needed to obtain the solutions. $Avg.Q_c$ is defined as

$$Avg.Q_c = \frac{\sum_{k \in C} Q_k}{|C|} \tag{26}$$

Similarly, the maximum absolute DIOB among all clusters, denoted by $Max.Q_c$, is defined as

$$Max.Q_c = max_{k \in C}\{Q_k\} \tag{27}$$

% Diff refers to the percentage difference between a metric obtained by MFUA and CPLEX divided by that found by CPLEX in Table 3 (by MFUA and K-means clustering divided by that found by K-means clustering in Table 4). A negative % Diff in CPU indicates that MFUA reduces the computational time. A positive % Diff in Q represents that the solution quality found by MFUA is worse than that found by CPLEX (in Table 3) or K-means clustering (in Table 4).

Table 3: Computational performances of MFUA and the branch-and-cut algorithm by CPLEX.

Hour	$\|C\|$	$\|S\|$	n^{min}	n^{max}	MFUA			CPLEX			Diff in	% Diff	
					$Avg.Q_c$	$Max.Q_c$	CPU(s)	$Avg.Q_c$	$Max.Q_c$	CPU(s)	$Avg.Q_c$	CPU	Q
07:00-08:00	2	20	10	20	0.00	0.00	0.01	0.00	0.00	3.26	0.00	-32500	0.00
	3	30			0.00	0.00	0.01	0.00	0.00	5.50	0.00	-54900	0.00
	2	40	20	30	0.50	1.00	0.02	0.50	1.00	8.95	0.00	-14817	0.00
	3	50			0.67	1.00	0.02	0.67	1.00	34.97	0.00	-49857	0.00
	3	60			0.00	0.00	0.03	0.00	0.00	73.34	0.00	-104671	0.00
	4	70			2.00	2.00	0.03	1.33	2.00	110.12	0.67	-122256	33.33
	4	80			2.75	3.00	0.04	1.25	2.00	184.34	1.50	-167482	54.55
	5	90			4.50	5.00	0.04	1.75	2.00	236.04	2.75	-196600	61.11
	5	100			3.40	4.00	0.04	2.00	4.00	298.53	1.40	-248675	41.18
	6	110			6.80	6.00	0.05	4.00	5.00	389.34	2.80	-138950	52.94
	6	120			4.83	5.00	0.05	4.20	5.00	423.45	0.63	-120886	27.59
	7	130			5.17	6.00	0.06	4.50	5.00	498.67	0.67	-127764	12.90
	7	140			5.67	6.00	0.08	4.00	4.00	589.34	1.67	-140219	29.41
	8	150			7.29	9.00	0.06	6.40	8.00	639.67	0.89	-138959	37.25
	8	160			7.14	8.00	0.07	5.86	8.00	789.78	1.29	-151781	18.00
	9	170			6.88	7.00	0.09	6.00	6.00	912.42	0.00	154547	12.50
	9	180			5.75	7.00	0.09	4.50	5.00	1146.78	1.25	-184865	21.74
17:00-18:00	2	20	10	20	0.00	0.00	0.01	0.00	0.00	2.30	0.00	-22900	0.00
	3	30			1.00	2.00	0.01	1.00	2.00	3.50	0.00	-34900	0.00
	2	40	20	30	0.50	1.00	0.02	0.50	1.00	11.30	0.00	-56400	0.00
	3	50			0.67	1.00	0.03	0.33	1.00	48.63	0.33	-16200	50.00
	3	60			0.00	0.00	0.03	0.00	0.00	68.25	0.00	-227400	0.00
	4	70			2.00	3.00	0.04	1.00	2.00	105.23	1.00	-262975	50.00
	4	80			1.00	2.00	0.04	1.00	2.00	196.47	0.00	-491075	0.00
	5	90			2.75	3.00	0.05	1.20	3.00	241.89	1.55	-483680	45.45
	5	100			2.80	4.00	0.05	2.75	4.00	305.26	0.05	-610420	21.43
	6	110			3.40	5.00	0.07	2.50	4.00	378.52	0.90	-540643	41.18
	6	120			2.33	5.00	0.06	0.60	4.00	435.89	1.73	-726383	57.14
	7	130			3.00	5.00	0.07	1.80	5.00	492.01	1.20	-702771	50.00
	7	140			2.67	4.00	0.08	1.00	5.00	521.49	1.67	-651763	25.00
	8	150			2.71	8.00	0.07	2.00	7.00	632.78	0.71	-903871	36.84
	8	160			3.13	7.00	0.08	2.57	7.00	801.62	0.55	-1001925	28.00
	9	170			3.13	6.00	0.09	2.38	6.00	925.26	0.75	-1027967	24.00
	9	180			2.88	5.00	0.09	2.50	4.00	1106.19	0.38	-1229000	13.04

Table 4: Computational performances of MFUA and K-means clustering for the instances constructed from the hour 17:00 - 18:00.

$\|C\|$	$\|S\|$	n^{min}	n^{max}	Kmeans			Diff in	% Diff	
				$Avg.Q_c$	$Max.Q_c$	CPU(s)	$Avg.Q_c$	CPU	Q
2	20	10	20	0.00	0.00	0.03	0.00	-200	0.00
3	30			3.33	5.00		-2.33	-200	-69.70
2	40	20	30	4.00	4.00	0.03	-2.00	-50	-87.50
3	50			4.00	6.00		-3.33	0	-83.25
3	60			10.67	16.00		-10.67	0	-100.00
4	70			14.00	18.00		-12.00	25	-85.71
4	80			14.50	16.00		-13.50	25	-93.10
5	90			12.80	31.00		-10.05	40	-78.52
5	100			18.60	26.00		-15.80	40	-84.95
6	110			25.00	61.00		-21.60	57	-86.40
6	120			30.67	74.00		-28.34	50	-92.40
7	130			21.71	83.00		-18.71	57	-86.18
7	140			32.86	88.00		-30.19	63	-91.87
8	150			26.75	88.00		-24.04	57	-91.12
8	160			30.75	90.00		-27.62	63	-89.84
9	170			28.67	115.00		-25.54	67	-90.31
9	180			28.89	126.00		-26.01	67	-91.15

The computational results are presented in Table 3 and Table 4. Key observations are as follows.

- In Table 3, the % Diff in Q ranged from 0.00% to 61.11% between MFUA and CPLEX. It was resulted from the small value of Q in most instances and, therefore, even a small difference could lead to a large % Diff. However, the differences between the average Q_c resulting from MFUA and CPLEX were at most 2.80, indicating that the communities identified by MFUA and an exact method were of comparable solution quality. Since optimality was guaranteed by CPLEX, the solutions obtained by MFUA were, therefore, demonstrated to be of good quality.

- In Table 3, DIOB in a cluster produced by MFUA was at most nine bike trips. This observation suggests that our solution methodology could effectively reduce the imbalance of bike flows between clusters.

- In Table 3, the solution time of CPLEX increased significantly as the problem size increased, from 2.30s for the smallest instance (20 stations) to 1146.78s for the largest one (180 stations). The % Diff between running times was huge, over ten thousand times in all instances. Although the % Diff between running times of MFUA and Kmeans was positive in most instances, the gap is not greater than 0.06s. In practice, BR operations are dynamic and may require timely updates on the plan due to various situations (e.g., changes in demands and repositioning truck breakdowns). In the experiments, MFUA could produce solutions within a second in all the instances, while the high solution quality was demonstrated.

- We also compare the solutions produced by MFUA and K-means clustering in Table 4. We observe that the % Diff in Q was always non-positive for all instances and ranged between -78.52% and -100% when $|S| \geq 40$. This suggests that our proposed MFUA is significantly more effective in balancing the numbers of bikes among clusters than the widely adopted clustering approach K-means clustering. While the CPU time resulting from MFUA was higher, the difference was negligible as computing times of both approaches were less than 0.1s.

The above observations suggest that MFUA is more appropriate for clustering bike stations for BR in real-world applications.

5.4 Comparison of Clusters in Different Periods

In this section, by analyzing the computational time and the characteristics of the clusters identified, we suggest the benefits of our methodology for BR. The results of the comparison between clusters in different hours exhibit users' travel patterns and verify that BSS is a typical short-distance transportation mode.

The key performance indicators considered in this experiment are as follows:

- The average percentage of DIOB of clusters ($APDIOB$)

$$APDIOB = \frac{1}{|C|}\sum_{c\in C}\frac{Q_c}{TOT_c}\cdot 100\% \tag{28}$$

  where TOT_c is the total number of bike trips from and to cluster c.

- The average in-cluster geographical distance between bike stations (AD)

$$AD = \frac{1}{|C|}\sum_{c\in C}\frac{\sum_{s,r\in c} d_{sr}}{n_c\cdot(n_c-1)} \tag{29}$$

  where d_{sr} is the distance between bike stations s and r, and n_c is the number of bike stations in cluster c.

Similarly, we define the maximum of average in-cluster distances of all the clusters, $Max.D$, as

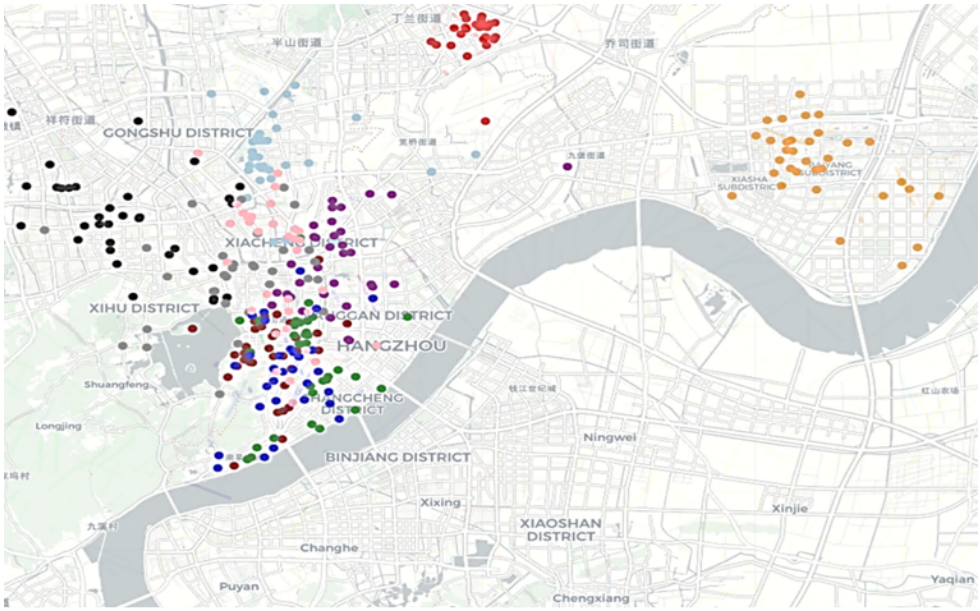
$$Max.D = max_{c\in C}\{\frac{\sum_{s,r\in c} d_{sr}}{n_c\cdot(n_c-1)}\} \tag{30}$$

Table 5: Characteristics of clusters in different periods.

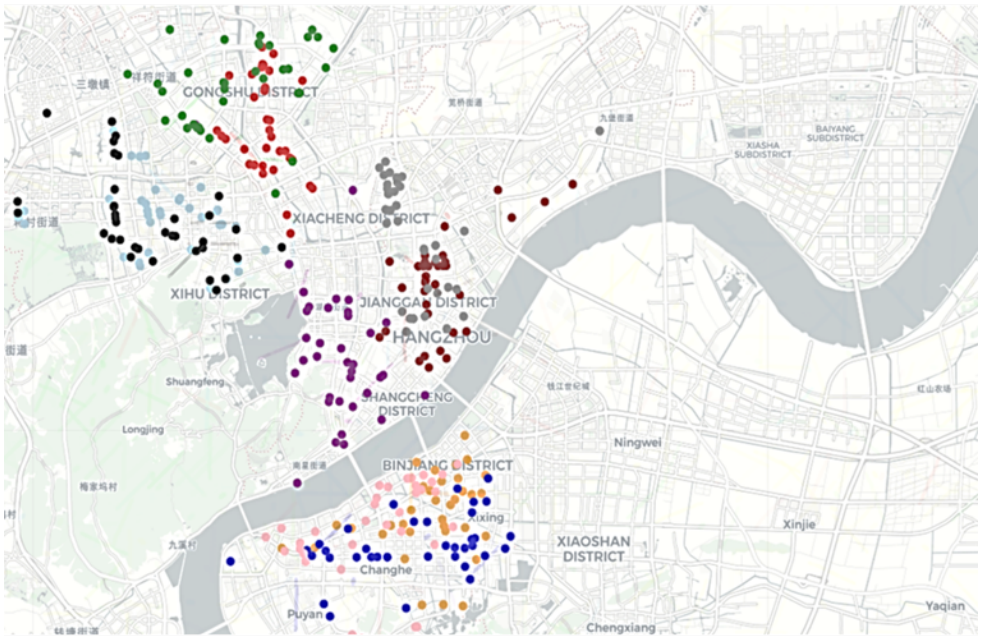
Instance	Q	$APDIOB$	AD(km)	$Max.D$(km)	$\|C\|$	$\|S\|$	CPU(s)
07:00-08:00	2548	6.45%	3.70	3.92	77	2471	0.197
08:00-09:00	1462	4.27%	3.00	3.84	75	2333	0.191
10:00-11:00	290	1.94%	2.61	3.02	72	2026	0.157
14:00-15:00	286	2.35%	2.25	3.57	68	1896	0.109
17:00-18:00	474	2.67%	3.19	3.75	74	2264	0.148
18:00-19:00	746	4.54%	3.23	3.15	73	2171	0.149

The computational results are summarized in Table 5. Some key observations are illustrated as follows:

- MFUA could produce solutions within a second for large-scale instances. The computational time for the largest instance (07:00-08:00) was only 0.197s. The results demonstrate that our methodology can be applied for practical instances.



(a) Ten example clusters in 07:00-08:00



(b) Ten example clusters in 14:00-15:00

Figure 5: Visualization of ten sample clusters in a peak hour and an off-peak hour.

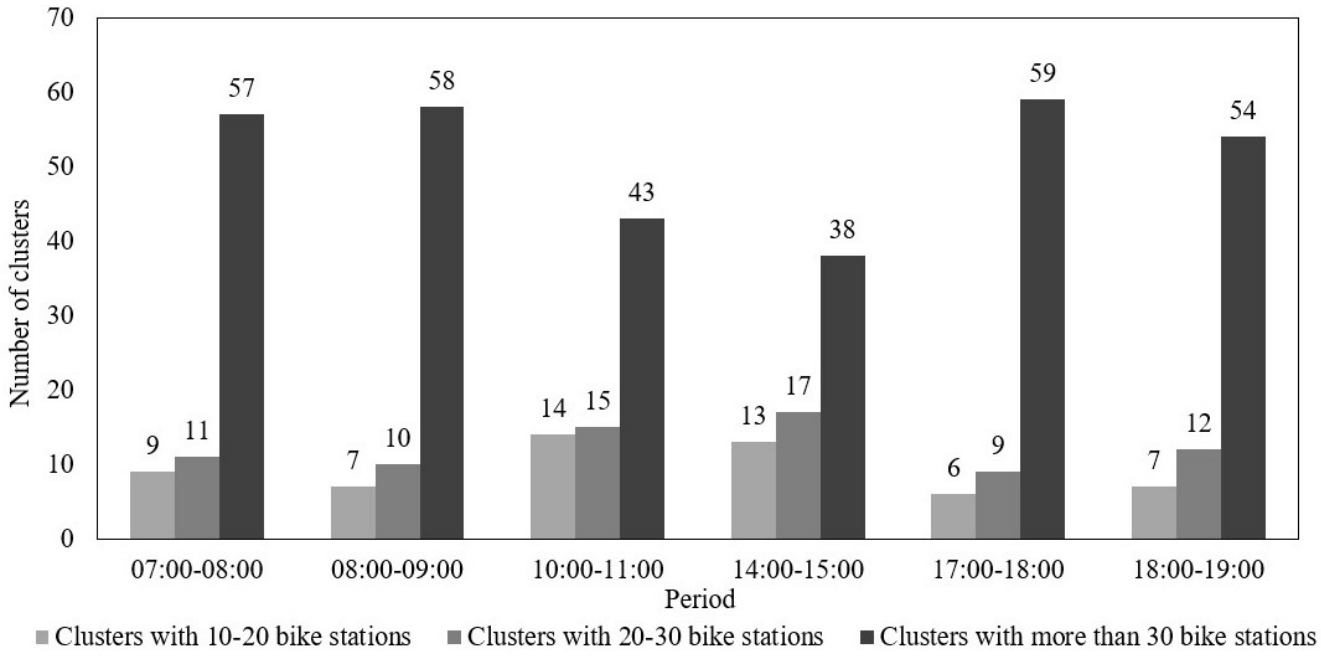


Figure 6: Numbers of clusters of various scales in different hours of the day.

- The small values of $APDIOB$ (within 6.45% in all instances) suggest that our methodology can provide good-quality solutions for large-scale problems. Such small values indicate that the clusters formed are relatively independent and can help avoid over-replenishing bikes and better match bikes to demands.

- Interestingly, our results exhibit that the bike stations in the same cluster are geographically close to each other (see AD in Table 5). We visualize ten clusters identified by our methodology during a peak hour (07:00-08:00) and an off-peak hour (14:00-15:00). Bike stations in the same cluster are of same color in Figure 5. We can attribute this phenomenon to the main characteristic of bike-sharing systems – short-distance transportation. BR within such a cluster can be more efficient because long-distance repositioning can be avoided. Our results suggest that, although geographic proximity highly impacts clustering, our demand-driven framework using trip records shall also be considered for effective clustering for BR in managing BSSs.

- The optimal number of clusters varies according to the hour of the day. In the morning and evening peak hours, users commute and, therefore, more bikes and bike stations are accessed. In off-peak hours, the origins and destinations are more likely those places near their working offices, such as convenience stores and restaurants. Similarly, the average distance between bike stations in the same

cluster during off-peak hours is also shorter than that during peak hours. Thus, shared-bike operators can set different numbers of repositioning trucks during different periods.

- We find that the percentage of large-size clusters (NBS$\geq$30) is greater than 70% in peak hours and 50% in off-peak hours (see in Figure 6). As a short-distance transportation mode, each originating bike station is typically associated with a limited number of destinations. Thus, clusters tend to be dense within a small region. Particularly, in peak hours, users tend to travel between residential areas and nearby public transport hubs.

We then examine the similarity between the clusters formed in different hours. The results are summarized in Table 6. The similarity is calculated as follows. We define $y_{sr}^{t_p}$ as

$$y_{sr}^{t_p} = \sum_{c \in C} x_s^{c,t_p} \cdot x_r^{c,t_p} \tag{31}$$

where x_s^{c,t_p} indicates if station s is in cluster c in period t_p.

The difference between clustering results in t_p and t_k is calculated by

$$Ed^{t_p,t_k} = \|Y_{t_p}^{t_p,t_k} - Y_{t_k}^{t_p,t_k}\|_2 \tag{32}$$

where a component of the matrix $Y_{t_p}^{t_p,t_k}$ indicates whether a pair of bike stations are in the same cluster in period t_p, which is indexed by $y_{sr}^{t_p}$. Note that the used bike stations in different hours may be different. $Y_{t_p}^{t_p,t_k}$ includes all the used bike stations in the periods t_p and t_k.

The maximum difference between clustering results in t_p and t_k is

$$Ed_{max}^{t_p,t_k} = \sqrt{\frac{n^{t_p,t_k} \cdot (n^{t_p,t_k} - 1)}{2}} \tag{33}$$

where n^{t_p,t_k} is the total number of used bike stations in periods t_p and t_k.

Thus, we define the similarity between clustering results in t_p and t_k as

$$\beta^{t_p t_k} = 1 - \frac{Ed^{t_p,t_k}}{Ed_{max}^{t_p,t_k}} \cdot 100\% \tag{34}$$

The observations from the comparison are as follows:

- It is observed that the clusters formed in morning peak hours and evening peak hours are similar. It

indicates that users' travel patterns are similar in the morning and evening. For instance, users ride from residential areas to public transport hubs in the morning and return the reverse way after work.

- Clusters formed in peak hours are similar. In peak hours, users typically commute between residential areas and work places (e.g., industrial parks and business districts). On the contrary, clusters of peak hours and off-peak hours are less similar due to the bike flow dynamics.

- Generally, the similarity between clusters in different hours is greater than 82% as travelers use shared-bikes mainly for short-distance trips.

Table 6: Similarity between clusters formed in different hours

	Hours	$\beta^{t_p t_k}$
Morning peak hour & Evening peak hour	07:00 & 17:00	84.84%
	08:00 & 17:00	84.19%
	07:00 & 18:00	84.68%
	08:00 & 18:00	84.08%
Morning peak hour & Morning peak hour	07:00 & 08:00	85.53%
Evening peak hour & Evening peak hour	17:00 & 18:00	85.69%
Off-peak hours	10:00 & 14:00	83.81%
Peak hour & Off-peak hour	07:00 & 10:00	82.89%
	08:00 & 10:00	82.30%
	17:00 & 14:00	82.63%
	18:00 & 14:00	83.32%

Below provides the key takeaways of our computational studies:

- MFUA is able to provide quick solutions to large-scale bike station clustering problems. It can provide solutions in less than 0.1 seconds in all our computational experiments where the number of bike stations is at most 180. On the other hand, state-of-the-art MIP solver CPLEX requires hundreds seconds in most of the instances. Since dynamic bike station clustering in practice requires fast solutions (e.g., in seconds), MFUA is more appropriate than CPLEX for implementation. While there is a tradeoff between computational speed and solution quality, the difference between the average DIOB is at most 2.80 in all instances.

- Our computational experiments suggest that solutions preserve geographic properties when clustering bike stations. The bike stations within a cluster are typically close to each other. This is due to the fact that shared bikes are usually used for last-mile transportation.

- Bike clustering solutions for peak hours are similar but appear to be less similar when compared with solutions for the off-peak hours. This is due to the commuting patterns of the users.

6 Conclusion

This paper proposes a novel demand-driven framework to cluster bike stations for bike repositioning, which is based on the optimal balancing of bike flows among different communities. An optimization model is developed for clustering with the objective to minimize the difference between the inflow and outflow of bikes to and from each community. A modified fast unfolding algorithm is proposed to solve the problem. Computational experiments based on the world's largest bike sharing system, Hangzhou public bike service system, were conducted to examine the performance of the proposed algorithm and to provide insights into bike station clustering. The comparison of clusters in different hours provides an understanding of bike user travel patterns.

Based on our proposed methodology and the real-world data on the bike-sharing system, our research has offered the following insights.

(i). The computational results show the performance of the proposed approach and the effectiveness of clustering by user demands as an alternative to forming clusters of bike stations. The proposed approach is shown to produce high-quality solutions in significantly less computational time for realistic applications.

(ii). Clusters found by our methodology provide a good operational foundation for bike repositioning in terms of the geographic proximity of bike stations in the same cluster, the balance of bike flows, and the number of bike stations in each cluster. The three factors are essential for dynamic bike repositioning.

(iii). Travelers use shared bikes mainly for short-distance trips. While bike flow patterns vary according to the hour of the day, the clusters formed by the methodology still preserve high similarity on the basis of flow balancing.

There are limitations of our research that shall be remarked. For example, as discussed in Section 3, the demands defined in this research are the bike trips actually realized. Unobserved demands in bike sharing systems have not been considered, which may be interesting to incorporate for further analysis. Potential

solutions to capture or estimate unobserved demands include the collection of bike request transactions via mobile apps or by statistical methods such as Tobit Model (Tobin, 1958). Moreover, our research considers station-based bike sharing systems. In a free-floating bike sharing system, there are no stations and it is not straightforward to construct a network/graph. New solution methods are required for such a system.

There are future research studies, which can be extended from this work.

(i). If real-time information about the users' trips are provided, would the optimal clustering decisions be different? If so, how well would the solutions be improved due to the real-time information? It would also be exciting to incorporate advanced demand prediction models (e.g., Huang et al., 2020) for more accurate information and better clustering decisions. Our approach could be applied in dynamic bike repositioning by incorporating real-time trip records. Based on real-time and predicted trips, the whole bike station network would be divided into different sub-networks hour by hour for better clustering decisions.

(ii). Our proposed methodology is not restricted to only for bike repositioning, but can be applied to other shared mobility systems, such as carsharing (Layla and Minner, 2021) or even multi-model transit networks (Li et al., 2020; Wu et al., 2020; Luo et al., 2021). It would be interesting to investigate the clustering decisions and optimal solution structures of these shared mobility systems, as the user demand and travel behaviors are expected to be quite different from those in bike sharing systems.

(iii). Bike repositioning at an operational level is typically considered and solved as vehicle routing problems. The bike station clustering and vehicle routing decisions may be jointly determined for more effective solutions. However, the computational complexity of solving such an integrated problem is expected to be high. Simplified, yet effective, mathematical models and efficient computational algorithms are required for practical implementation. It would be necessary to develop further solution approaches, including exact approaches and combinations of algorithms proposed in the domain of network theory and metaheuristics. It would be useful to solve network-based problems with the guarantee of solution quality.

Acknowledgment

This research was partially supported by the Germany/Hong Kong Joint Research Scheme Germany Academic Exchange Service (DAAD) and Research Grants Council (RGC) of Hong Kong (G-HKU703/19),

General Research Fund (GRF), Research Grants Council (RGC) of Hong Kong (grant 17200820), and the 2019 Guangdong Special Support Talent Program—Innovation and Entrepreneurship Leading Team (China) (2019BT02S593), and .

References

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

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

[3] Bay Wheels, 2021. Members get unlimited 45-minute rides all year long. https://www.lyft.com/bikes/bay-wheels/pricing, Accessed on December 6, 2021.

[4] Blondel, V.D., Guillaume, J.-L., Lambiotte, R., Lefebvre, E., 2008. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10), P10008.

[5] Chemla, D., Meunier, F., Calvo, R.W., 2013. Bike sharing systems: Solving the static rebalancing problem. *Discrete Optimization*, 10, 120–146.

[6] Chang, C.S., Lee, D.S., Liou, L.H., Lu, S.M., Wu, M.H., 2017. A probabilistic framework for structural analysis and community detection in directed networks. *IEEE/ACM Transactions on Networking*, 26(1), 31–46.

[7] Citibike, 2021. Membership plans. https://account.citibikenyc.com/access-plans, Accessed on December 6, 2021.

[8] Dell'Amico, M., Hadjicostantinou, E., Iori, M., Novellani, S., 2014. The bike sharing rebalancing problem: Mathematical formulations and benchmark instances. *Omega*, 45, 7–19.

[9] Dell'Amico, M., Iori, M., Novellani, S., Subramanian, A., 2018. The bike sharing rebalancing problem with stochastic demands. *Transportation Research Part B: Methodological*, 118, 362–380.

[10] DeMaio, P., 2009. Bike-sharing history impacts models of provision and future. *Journal of Public Transportation*, 12(4), 41–56.

[11] Du, W.-B., Zhang, M.-Y., Zhang, Y., Cao, X.-B., Zhang, J., 2018. Delay causality network in air transport systems. *Transportation Research Part E: Logistics and Transportation Review*, 118, 466–476.

[12] Du, Y., Deng, F., Liao, F., 2019. A model framework for discovering the spatio-temporal usage patterns of public free-floating bike-sharing system. *Transportation Research Part C: Emerging Technologies*, 103, 39–55.

[13] Erdoğan, G., Battarra, M., Wolfler Calvo, R., 2015. An exact algorithm for the static rebalancing problem arising in bicycle sharing systems. *European Journal of Operational Research*, 245(3), 667–679.

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

[15] Fortunato, S., 2010. Community detection in graphs. *Physics Reports*, 486(3-5), 75–174.

[16] Froehlich, J., Neumann, J., Oliver, N., 2009. Sensing and predicting the pulse of the city through shared bicycling. In:*Proceedings of the 21st international jont conference on Artifical intelligence*, San Francisco, CA, USA, 1420–1426.

[17] Garey, M.R., Johnson, D.S., Stockmeyer, L., 1976. Some simplified NP-complete graph problems. *Theoretical Computer Science*, 1(3), 237–267.

[18] Gervini, D., Khanal, M., 2019. Exploring patterns of demand in bike sharing systems via replicated point process models. *Journal of the Royal Statistical Society Series C*, 68(3), 585–602.

[19] Haider, Z., Nikolaev, A., Kang, J.E., Kwon, C., 2018. Inventory rebalancing through pricing in public bike sharing systems. *European Journal of Operational Research*, 270(1), 103–117.

[20] Hasija, S., Shen, Z.-J.M., Teo, C.-P., 2020. Smart city operations: Modeling challenges and opportunities. *Manufacturing & Service Operations Management*, 22(1), 203–213.

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

[22] Huang, D., Chen, X., Liu, Z., Lyu, C., Wang, S., Chen, X. 2020. A static bike repositioning model in a hub-and-spoke network framework. *Transportation Research Part E: Logistics and Transportation Review*, 141, 102031.

[23] Kabra, A., Belavina, E., Girotra, K., 2019. Bike-share systems: Accessibility and availability. *Management Science*, 66(9), 3803-3824.

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

[25] Legros, B., 2019. Dynamic repositioning strategy in a bike-sharing system; how to prioritize and how to rebalance a bike station. *European Journal of Operational Research*, 272(2), 740–753.

[26] Li, Y., Szeto, W.Y., Long, J., Shui, C.S., 2016. A multiple type bike repositioning problem. *Transportation Research Part B: Methodological*, 90, 263–278.

[27] Luo, X., Gu, W., Fan, W., 2021. Joint design of shared-bike and transit services in corridors. *Transportation Research Part C: Emerging Technologies*, 132, 103366.

[28] Li, X., Luo, Y., Wang, T., Jia, P., Kuang, H. 2020. An integrated approach for optimizing bi-modal transit networks fed by shared bikes. *Transportation Research Part E: Logistics and Transportation Review*, 141, 102016.

[29] Locobike, 2021. FAQ. https://loco.hk/bike/faq, Accessed on December 6, 2021.

[30] Martin, L., Minner, S. 2021. Feature-based selection of carsharing relocation modes. *Transportation Research Part E: Logistics and Transportation Review*, 149, 102270.

[31] Meituan Bike, 2021. Mobike. https://mobike.com/, Accessed on December 6, 2021.

[32] Mesa-Arango, R., Ukkusuri, S.V., 2015. Demand clustering in freight logistics networks. *Transportation Research Part E: Logistics and Transportation Review*, 81, 36–51.

[33] Metropolradruhr, 2021. Prices. https://www.metropolradruhr.de/en/prices/, Accessed on December 6, 2021.

[34] Negahban, A., 2019. Simulation-based estimation of the real demand in bike-sharing systems in the presence of censoring. *European Journal of Operational Research*, 277(1), 317–332.

[35] Newman, M.E.J., 2004. Fast algorithm for detecting community structure in networks. *Physical Review E*, 69(6), 066133.

[36] Pal, A., Zhang, Y., 2017. Free-floating bike sharing: Solving real-life large-scale static rebalancing problems. *Transportation Research Part C: Emerging Technologies*, 80, 92–116.

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

[38] Raviv, T., Tzur, M., Forma, I.A., 2013. Static repositioning in a bike-sharing system: models and solution approaches. *EURO Journal on Transportation and Logistics*, 2(3), 187–229.

[39] Santander Cycles, 2021. Hire a Santander Cycle in London. https://tfl.gov.uk/modes/cycling/santander-cycles, Accessed on December 6, 2021.

[40] Schuijbroek, J., Hampshire, R.C., Hoeve, W.J.v., 2017. Inventory rebalancing and vehicle routing in bike sharing systems. *European Journal of Operational Research*, 257(3), 992–1004.

[41] Shaheen, S.A., Zhang, H., Martin, E., Guzman, S., 2011. China's Hangzhou public bicycle. *Transportation Research Record: Journal of the Transportation Research Board*, 2247(1), 33–41.

[42] Shui, C.S., Szeto, W.Y., 2020. A review of bicycle-sharing service planning problems. *Transportation Research Part C: Emerging Technologies*, 117.102648

[43] Sun, X., Tang, W., Chen, J., Zhang, J., 2020. Optimal investment strategy of a free-floating sharing platform. *Transportation Research Part E: Logistics and Transportation Review*, 138, 101958.

[44] Szeto, W.Y., Shui, C.S., 2018. Exact loading and unloading strategies for the static multi-vehicle bike repositioning problem. *Transportation Research Part B: Methodological*, 109, 176–211.

[45] Tobin, J. 1958. Estimation of relationships for limited dependent variables. *Econometrica*, 26(1), 24–36.

[46] Van Nguyen, T., Zhang, J., Zhou, L., Meng, M., He, Y., 2019. A data-driven optimization of large-scale dry port location using the hybrid approach of data mining and complex network theory. *Transportation Research Part E: Logistics and Transportation Review*, 134, 101816.

[47] Wu, L., Gu, W., Fan, W., Cassidy, M.J., 2020. Optimal design of transit networks fed by shared bikes. *Transportation Research Part B: Methodological*, 131, 63–83.

[48] Yang, Z., Chen, J., Hu, J., Shu, Y., Cheng, P., 2019. Mobility modeling and data-driven closed-loop prediction in bike-sharing systems. *IEEE Transactions on Intelligent Transportation Systems*, 20(12), 1–12.

[49] Zhang, D., Yu, C., Desai, J., Lau, H.Y.K., Srivathsan, S., 2017. A time-space network flow approach to dynamic repositioning in bicycle sharing systems. *Transportation Research Part B: Methodological*, 103, 188–207.

[50] Zhang, J., Meng, M., 2019. Bike allocation strategies in a competitive dockless bike sharing market. *Journal of Cleaner Production*, 233, 869–879.

[51] Zhou, M.-Y., Cai, S.-M., Fu, Z.-Q., 2012. Traffic dynamics in scale-free networks with tunable strength of community structure. *Physica A: Statistical Mechanics and Its Applications*, 391(4), 1887–1893.

[52] Zhou, X., 2015. Understanding spatiotemporal patterns of biking behavior by analyzing massive bike sharing data in Chicago. *PLoS One*, 10, 1–20.

[53] Zhu, Y., Diao, M., 2020. Understanding the spatiotemporal patterns of public bicycle usage: A case study of Hangzhou, China. *International Journal of Sustainable Transportation*, 14, 163–176.