J Vis (2019) 22:587–607
https://doi.org/10.1007/s12650-019-00557-6

REGULAR PAPER

Seonghoon Ban · Kyung Hoon Hyun

Curvature-based distribution algorithm: rebalancing bike sharing system with agent-based simulation

Received: 20 August 2018 / Revised: 13 March 2019 / Accepted: 5 April 2019 / Published online: 17 April 2019

Abstract The selected rebalancing strategy determines the service results for bike sharing system (BSS). All cities have different operators, such as the number of stations, bikes, and rebalancing trucks, traffic congestion patterns, terrain maps, and maintenance costs. Therefore, each city needs a rebalancing strategy of its own. However, identifying an appropriate rebalancing strategy requires a method that can simulate a large volume of rebalancing operators. In this paper, we propose a novel method to dynamically rebalance BSS with agent-based simulation. We developed a curvature-based distribution algorithm which allows simulations with a large number of rebalancing trucks and stations (40 trucks and 581 stations), compared to previous studies on bike rebalancing. The algorithm uses a three-dimensional (3D) terrain map created with a large volume of BSS usage data (3,595,383 observations) to generate dynamic truck routes for agent-based simulation. In addition, traffic congestion data were used as weighting values to improve the accuracy of the proposed method for truck route generation. Thus, the proposed algorithm can test different rebalancing strategies that are suitable for a specific city with various spatiotemporal conditions. Specifically, the system analyst can simulate different truck numbers, working ranges, and working hours to identify a suitable strategy for various cities to estimate yearly budget. The algorithm is simple to implement and adaptive to various bike usage data. The research also proposes a visual analysis method based on simulated results of rebalance imbalance metrics, service failure rate, number of daily operations, daily truck travel distance, and stochastic bike usage behavior.

Keywords Bicycle-sharing system · Agent-based simulation · Virtual environment · Rebalancing problem · Urban traffic · Rebalancing strategy

1 Introduction

Bike sharing systems (BSSs) have been standardized in large metropolitan cities in America, Europe, and Asia (de Chardon et al. 2016; Do and Noh 2014; O'Brien 2013, 2017). The BSS provides shared bicycles for individuals for a relatively short-term period within an urbanscape. Instead of driving motorized vehicles, individuals can easily rent and return the bicycles at bike stations that are placed in multiple locations in the city. The BSS is recognized as a successful and effective service that can help to improve urban traffic and is also environmentally friendly (Dötterl et al. 2017; Pal and Zhang 2017). A key aspect of the BSS is to make the service available to its users throughout the city (Conticelli et al. 2018). This is also known as a bike rebalancing problem. Stabilizing bike exhaustion and congestion at the stations is especially critical for the BSS to maximize consistent service availability. For example, service users living near station A tend to use

S. Ban · K. H. Hyun (✉)
Hanyang University, Seoul 04763, Republic of Korea
E-mail: hoonhello@hanyang.ac.kr
Tel.: + 82-10-2811-1521

the BSS in the morning to commute to work near station B. There are no bikes left at station A for potential users; however, an ample number of bikes are left at station B. BSS around the world solves such problems by utilizing rebalancing trucks to pick up bikes from congested stations and redistribute them to stations where they are needed, thereby increasing service availability to potential users. The BSS rebalancing problem requires high computational costs to derive optimized solutions by considering multiple factors, such as the bike usage patterns per date and time, number of operable trucks, distances between stations, and possible traffic congestion depending on the time of day. Two methods address this problem: (1) visual analysis approach and (2) vehicle routing optimization approach. The visual analysis approach visualizes the bike usage data for the system operator to heuristically understand the bike usage status rather than provide analyzed information. Yan et al. (2018) designed a visual analysis system to evaluate bike usage patterns in different cities. They utilized spatial, temporal, and demographic information to analyze bike usage patterns. O'Brien (2013) created the Bike Share Map, a Web-based visualization system visualizing real-time bike usage of over 300 cities worldwide. The system operators and the truck drivers relocate bikes reactive to the real-time visualization of the station inventory. However, rebalancing bikes through data visualization makes it difficult to distribute bikes across the city. For example, a station with a shortage of bikes can be near stations with ample bikes; therefore, system users can walk to the nearest station to rent a bike. Thus, it is important to distribute the bikes throughout the city and increase overall bike accessibility. Unlike the visual analysis approach, the vehicle routing optimization approach attempts to compute the near-optimal solution for the rebalancing problem. Chemla et al. (2013) proposed a branch-and-cut algorithm for solving the rebalancing problem from a theoretical point of view. They used a maximum of 100 nodes and succeeded in finding the optimal solution nearest to the target state. Conversely, Regue and Recker (2014) used actual bike usage data for simulation. They proposed a proactive vehicle routing framework to solve the dynamic bike rebalancing problem. Their framework consists of a bike demand forecasting model, station inventory model, bike redistribution model, and a truck relocation model. Their model allows for proactively responding to bike demand rather than reactively relocating bikes. According to de Chardon et al. (2016), the rebalancing strategy determines the outcome of the service. All cities have a different number of stations, bikes, rebalancing trucks, traffic congestion patterns, service participants, and terrain map; therefore, each city needs to formulate a rebalancing strategy of its own. Small cities, for example, can utilize trucks with a large operating radius such that trucks can pick up and deliver across the city. Conversely, large cities can distribute trucks to each of the city districts operating within the predefined area. Further, the rebalancing truck operation strategy can be adjusted to a particular city budget. The system operator needs to consider the number of trucks, truck drivers, and working hours. Thus, the cities can benefit from truck relocation strategies more suitable for their conditions. Developing an efficient truck operation strategy requires an adjustable simulation system to test for the most optimized strategy suitable to a specific city. To do that, an agent-based simulation is important to test the interactions among spatiotemporal information and the trucks. Because the trucks already behave in predefined rule sets, it is essential to evaluate any macroscopic phenomenon that occurs when an agent depicting a truck behaves according to set rules.

In this paper, we propose a novel method to simulate various BSS rebalancing strategies. We developed a curvature-based distribution algorithm that uses the three-dimensional (3D) terrain map generated with the actual BSS usage data to conduct agent-based simulation. The 3D terrain map is used to define the rebalancing truck routes based on the curvatures created on the terrain map. The concave surface (crater) represents the shortage of bike availability in neighboring stations, and the convex plane (hill) represents the abundance of bike availability in neighboring stations. Moving the total number of bicycles from the local maxima to the local minima is consistent with the perspective of stability theory. The rebalancing truck can reach the neutral state by moving the bikes from hill to crater which reduces potential energy. To improve the accuracy of the simulation, the traffic congestion data were used as weighting values for truck route simulation. In this respect, it is possible to simulate rebalancing strategies for the BSS for different cities, while monitoring the visualized graphics for the truck operators. The algorithm allows finding the most suitable strategies depending on the size of the city, maintenance cost, number of trucks, and operational durations. To accomplish this, we conducted four primary tasks: (1) large volume (3,595,383 observations) of bicycle usage data were collected, and large volume of traffic congestion data were collected (1,499,126 observations); (2) 3D terrain map was created based on urban geographic data, BSS usage data, and traffic congestion data; (3) agent-based simulation was conducted within the terrain map; (4) impact of the proposed algorithm and optimal truck operation strategy was tested through validation experiments. The contribution of the study is as follows:

1. The research proposes a novel algorithm to dynamically rebalance the BSS utilizing agent-based simulation.
2. The proposed algorithm can test different rebalancing strategies that are suitable for a specific city with various spatiotemporal conditions.
3. The algorithm is simple to implement and adaptive to various bike usage data.
4. The research also proposes a visual analysis method on the simulated results of rebalance imbalance metric, service failure, number of daily operations, daily truck trip distance, and stochastic bike usage.

2 Related work

2.1 Data driven visual analysis

Data visualization is advantageous for understanding information and insight extraction. Data visualization helps to extract hidden insights by presenting the data in human-friendly graphics. According to Tufte (2011), data visualization helps to extract insights that are difficult to be perceived through statistical analysis. By visualizing data, it is possible to check the correlations and tendencies among variables instantly. Data visualization is especially useful in geographic and urban context. Data visualization researchers often create data maps when extracting insights from spatial and sociological data. John Snow, for example, marked the location of water pumps and the location of dead patients in London to identify the source of cholera (Tufte 2011), and Koblin (2009) visualized airline traffic over US territory to find airplane navigation patterns throughout a time sequence. Lee et al. (2018) visualized electronic waste export trajectories to track how e-wastes are broken down and shipped throughout the world, allowing researchers to understand the movement of hazardous e-waste better. The data map enables an effective evaluation of the relationships among variables. Another advantageous quality of the data map is that the graphical representation of geographic and social data can be multilayered. Figure 1 shows the data map of populations with cancer deaths throughout US counties. Each map portrays total of 21,000 numbers in 3056 counties and allows visual analysis of the difference in death rates from cancer by regions, gender, and cancer types (Tufte 2011). Data map visualization allows the viewer to instantly and intuitively dive into extracting insights from the data.

Data visualization is also effective and efficient in dealing with spatiotemporal information. According to Andrienko et al. (2003), we can construct two major questions from the three key components of spatiotemporal visualization (when, where, and what): (1) when $\rightarrow$ what + where; (2) what + where $\rightarrow$ when. The former is to describe the object(s) in a given location(s) based on the time or set of times. The latter is to describe the time or set of times based on the object(s) in a given location(s). They also introduced the concept of elementary and general search levels to distinguish the questions for different search targets. Elementary search level refers to describing the characteristics of the components, and the general search level refers to the dynamics of the components. For example, elementary "when" and elementary "what + where" describe the characteristics of object(s) and location(s) at a given time, while general "when" and general "what + where" describe the dynamics of overall situations over time. Researchers effectively utilized different search levels to better identify and compare information through data driven visualizations. For example, Xie et al. (2016) created a Web-based interactive visualization tool for rapid tsunami evacuation planning. The system allows users to simulate evacuation plans based on a dynamic and interactive map where users can select different search targets. Therefore, disaster managers and operators visually analyze spatiotemporal information for optimized decision-making. Pei et al. (2018) proposed a visual analysis system called BVis to enhance the manageability of traffic and bus station congestion patterns. In their works, the operators can check the spatial dynamics of traffic in regions as well as the characteristics of a specific road. Their system allows the operators to evaluate station and bus performance depending on traffic congestion using visualization. Yan et al. (2018) developed a visual analysis system to capture BSS usage patterns in/between cities. They employed a tensor factorization based on spatial, temporal, and user information to extract bike usage patterns. Their visualization allows operators to check the spatial distribution of bike storage in regions, as well as the specific number of stored bikes per stations. Thus, BSS researchers can visually analyze general regional differences as well as specific stations. O'Brien (2013) developed the Bike Share Map, an online visualization platform for the shared bicycle system of over 300 cities worldwide. The Bike Share Map uses real-time bike usage data and visualizes the availability of each bike station. Zhang et al. (2016) created a platform that visualizes bike usage patterns as

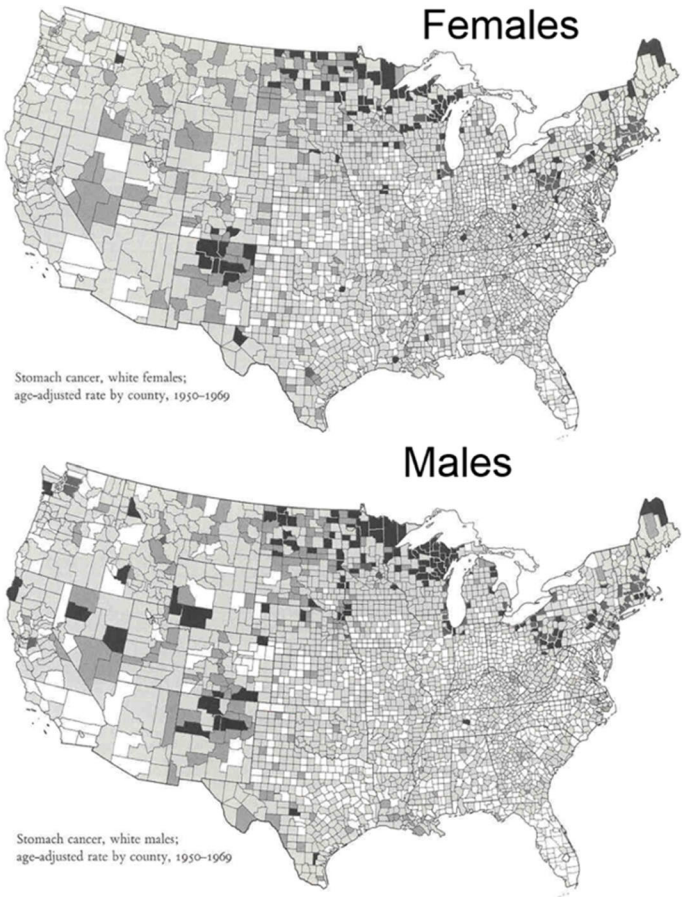

Fig. 1 Data map example (Tufte 2011)

well as bike storage rate dynamically. The literature on the visualization of spatiotemporal information focuses on improving comprehension of the large volume of data by transforming the data into multivariate graphical representations. Thus, various search levels of the spatiotemporal visualization allow analysts to relate or compare the elements of data to optimize a decision-making process. In this research, we further considered implementing agent-based simulation in addition to data visualization to enhance the capabilities of the visual analysis. We also developed a visual analysis system to help BSS operators manage the service.

2.2 Agent-based simulation

Agent-based simulation research is widely investigated in the field of architecture and urban planning. An agent-based simulation helps to evaluate complex and systematic problems. It uses software robot as system component and simulates interactions among the system components. Hillier and Hanson (1984) first introduced the space syntax to research spatial logic based on social relationships. The space syntax research enabled urban design researchers and practitioners to evaluate how people view, behave, and interact in an urbanscape (Hyun et al. 2016). Space syntax provides a foundation for the agent-based simulation to evaluate traffic distribution. Turner et al. (2001) provided the concept of a visibility graph for agent-based simulation. The visibility graph is created with isovists, and an isovist is also known as a viewshed that is used to measure the distances among spaces at where a person stands (Batty 2001). After the isovists are marked in the spaces in the grids, the visibility map is created by measuring distances between isovists (Fig. 2).

The visibility graph-based simulation simulates traffic distribution under the assumption that people take the path with the furthest isovist distance when exploring a new environment (Turner and Penn 1999). Next, the number of agents walking through each isovist is counted to evaluate the traffic distribution. In this respect, architects and urban planning designers can make effective wayfinding design decisions by

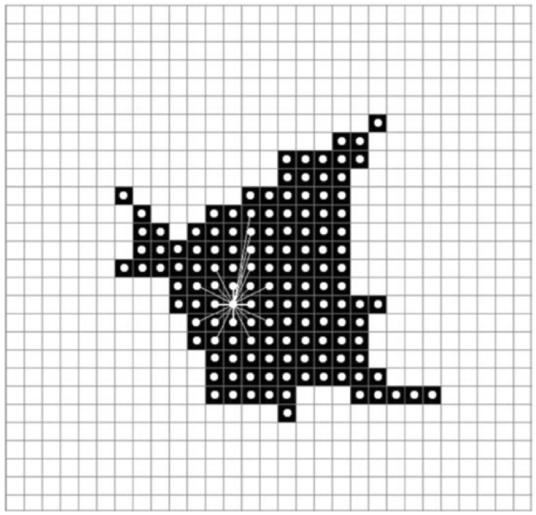

Fig. 2 Isovist map example (Batty 2001)

evaluating and optimizing traffic distribution. Min et al. (2017) used agent-based simulation to optimize visitor traffic in theme parks. By analyzing visitor traffic, theme park service facilities, including attractions, shops, and restaurants, can be positioned for optimal traffic distribution. Tang and Hu (2017) tested an agent-based simulation to evaluate crowd behavior for various design and planning scenarios. They evaluated crowd flow, travel time, population density, and public accessibility to improve traffic congestion and flow efficiency. Tang (2018) proposed the Avatar Agent virtual reality system which implemented immersive virtual reality and agent-based simulation with space syntax to analyze wayfinding behavior. The system allows capturing of user movement behavior to examine its correlation with demographic information, spatial recognition, and architectural elements. Yuksel (2018) introduced an agent-based evacuation simulation model to understand pedestrian traffic patterns based on artificial neural networks and genetic algorithm. His research successfully enabled agents to find building exits while considering targets, communications, and obstacles. Despite the significance of the research on agent-based simulation, most of the research concentrates on the wayfinding behavior, not the agents rebalancing supplies based on demands. In this respect, this research proposes a novel method that consists of visual analysis and agent-based simulation for rebalancing BSS.

2.3 Rebalancing bike sharing system

Rebalancing problems in BSS is an active research area; there are two major research segments: operator based and user participation based (Pal and Zhang 2017). The former utilizes trucks to rebalance the bikes while the latter redistributes bikes based on user participations. The operator-based segment can be more effective in rebalancing station-based BSS (Chiariotti et al. 2018) while the user participation-based segment can be more effective for the dockless BSS (Pan et al. 2018). However, both segments share static and dynamic rebalancing strategies. Static rebalancing strategy is used to relocate bicycles at prescribed time. Rainer-Harbach et al. (2013) proposed the rebalancing method based on variable neighborhood search and greedy heuristic for truck route generation. Their method statically rebalances bikes at a specific time. However, unlike the static rebalancing strategy, the dynamic rebalancing strategy is a strategy for the operator to relocate in real time. Although dynamic rebalancing is difficult to implement in the actual service operation, it can respond more effectively to bike imbalance than the static strategy. Chiariotti et al. (2018), for example, proposed a method to dynamically rebalance New York City's Citibike service. They used four-year-long BSS usage data and identified that dynamic rebalancing outperforms static rebalancing in service quality and maintenance cost. In addition, the dynamic rebalancing method can efficiently respond to rush hour traffic and seasonal variations. Kloimüllner et al. (2014) expanded their previous static rebalancing method by modifying it into a dynamic rebalancing method (Rainer-Harbach et al. 2013). They utilized greedy randomized adaptive search and variable neighborhood search to dynamically balance the BSS. An interesting point identified during their dynamic rebalancing research was to consider neighbor stations as either full or empty. In view of this factor, Dötterl et al. (2017) stressed the importance of

situational awareness for bike rebalancing. Although the authors concentrated on user participation-based rebalancing, they introduced the concept of neighboring station agents for proximity detection to better respond to bike imbalance and incentive pricing. Based on the literature review, we identified three important factors for bike rebalancing algorithms. First, response to real-time change is an important element for the truck routing algorithm. It is important to respond to a status change at the station and dynamically reroute the truck's path. For example, truck C plans to rebalance bikes at station A. If station A is already rebalanced either by truck B or through BSS user, it is no longer efficient for truck C to visit station A. Dynamic rebalancing methods can partially solve this problem. The current dynamic rebalancing methods are network based, and each time the truck arrives at the station, the new path is recalculated according to the change in bike storage at stations (Kloimüllner et al. 2014; Chiariotti et al. 2018; Regue and Recker 2014). However, these methods route the truck when the truck is at the station; in this manner, it will not reroute immediately when the situation changes while the truck is moving. This is done so that one can essentially flexibly respond to real-time change since bike storage rate can rapidly change while the rebalancing truck is on the move during rush hour. Thus, the consideration of real-time route changing will result in a more accurate rebalancing simulation. Second, regional imbalance calculation helps to dynamically respond to stochastic spatiotemporal conditions. As Andrienko et al. (2003) suggested, both elementary (bike storage rate of a specific station) and general (bike storage rate of regional stations) search levels are important. For example, the algorithm can concentrate on rebalancing bikes for the busiest stations during rush hour while distributing bikes among the neighboring stations in the afternoon. To our knowledge, the literatures on rebalancing operator-based BSS only utilize elementary searches. Finally, simulation utilizing a large number of trucks and stations is critical. As the BSS expands over time, the number of rebalancing trucks and stations increases (London utilizes 21 trucks and 725 stations; Seoul has 36 trucks and 1163 stations). In the case of network-based rebalancing truck algorithms, the computational cost increases dramatically depending on the number of stations. Moreover, if a considerable number of trucks are used, the relationship between trucks must be taken into consideration, and the computation involved will further increase. The optimization of these operations will become more important as the size of the BSS increases. Thus, it is important that the rebalancing algorithm be able to handle a large number of operators with a low computational cost. As illustrated in Table 1, the proposed method can dynamically rebalance BSS while responding to real-time route change and calculating regional imbalance. In case of responding to a real-time route change, we classified "considered," when the truck route is being recalculated instantly according to the station status change in real time, and "partially considered" for considering the station status change in real time but not rerouting instantly. The regional imbalance was classified as "considered" if the regional imbalance was included in the truck route calculation. In addition, the proposed method can simulate using a greater number of trucks and stations than reported in the previous

Table 1 Summary of the operator-based rebalancing method literatures

Research article	Type	Number of trucks	Number of stations	Responding to real-time route change	Regional imbalance calculation	Methodology
Rainer-Harbach et al. (2013)	Static	1–5	30–90	–	Not considered	Variable neighborhood search, greedy heuristic
Chemla et al. (2013)	Static	1	100	–	Not considered	Tabu search
Kloimüllner et al. (2014)	Dynamic	1–5	30–90	Partially considered	Not considered	Variable neighborhood search, greedy randomized adaptive search procedure
Chiariotti et al. (2018)	Dynamic	1	280	Partially considered	Not considered	Birth–death process
Regue and Recker (2014)	Dynamic	1–5	61	Partially considered	Not considered	Demand forecasting model; station inventory model; redistribution needs model; vehicle routing model
The proposed method	Dynamic	10–40	581	Considered	Considered	Curvature-based distribution algorithm, agent-based simulation

literatures. We tested 10, 20, 30, and 40 trucks; however, this number can be easily increased owing to the method's agent-based approach.

3 Methods

This paper proposes a curvature-based distribution algorithm that rebalances BSS through agent-based simulation. To implement the algorithm, a 3D terrain map is constructed as shown in Fig. 3. The 3D terrain map is created with BSS usage data, historical traffic congestion data, and an illustrated satellite image of the city. The shape of the 3D terrain map changes corresponding to the bike usage data, hence generating bike usage data in a time sequence. Next, the agent-based simulation is conducted on the 3D terrain map. As shown in Fig. 3c, the sphere is the agent (truck); the agent will roll down the gradient in a direction from high to low. The traffic congestion data are used as the weighting value for the moving speed of the distribution truck. The proposed algorithm allows extracting two significant insights. First, it allows visualizing bike usage patterns in each district and in each station on weekdays and weekends during four different seasons. Second, the agent-based simulation provides efficient truck navigation plans based on the 3D terrain map. The proposed algorithm is developed and implemented with Java.

3.1 Data structure

The proposed algorithm uses two large datasets: (1) bike usage data from the BSS and (2) historical traffic congestion data (Table 2). The BSS dataset consists of bike usage patterns, including unique trip identifier, trip start time, trip stop time, bike identifier, trip duration, departing station identifier, destination station identifier, station name, and demographic information, such as user type (service subscriber or guest user), user gender, and user birth year. The historical traffic congestion data consist of time (date and time corresponding to congestion level), region identified (unique arbitrary number to represent each region), bus count (the number of buses used to estimate traffic), number of reads (number of GPS probes received (or used) for estimating the speed for that segment), and speed (estimated congestion level in miles per hour).

3.2 3D terrain map and agent-based simulation

The 3D terrain map was created for the curvature-based distribution algorithm to conduct agent-based simulation. The 3D terrain map is created based on the longitudinal and latitudinal location of bike stations, and the number of surplus and insufficient bikes in the station. The longitude and latitude locations of stations were converted into x and y coordinates in the Cartesian plane. The grid is created on the x and y planes. The summation of surplus and insufficient number of bikes per station within designated distances between intersections of the grid is expressed in the z coordinates that define the height of the grid in the 3D

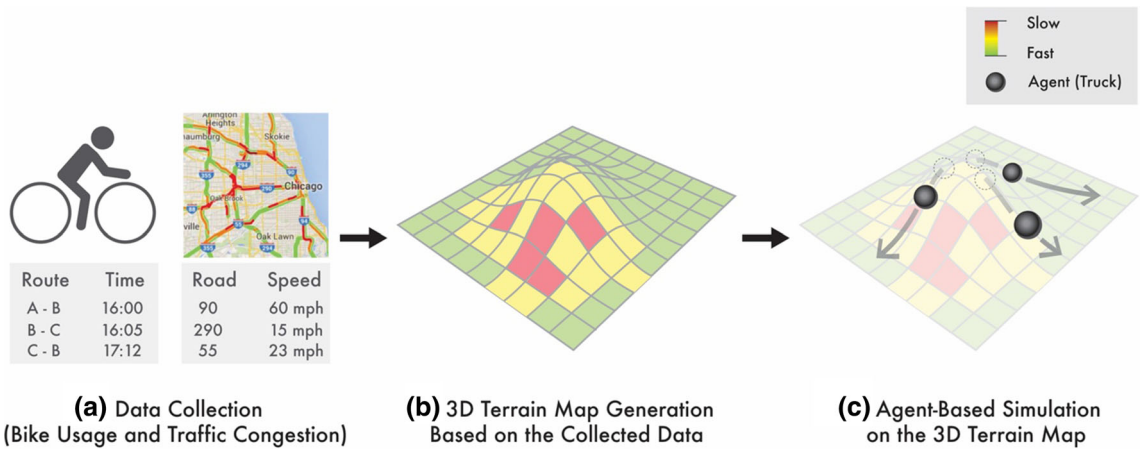

Fig. 3 Curvature-based distribution algorithm overview **a** data collection; **b** 3D terrain map generation, and **c** agent-based simulation

Table 2 Data scheme of the BSS and traffic congestion in Chicago

BSS datasets			Historical traffic congestion datasets		
Data name	Data type	Data sample	Data name	Data type	Data sample
Trip_id	ID	9074846	Time	Date and time	3/31/16 23:50
Starttime	Date and time	3/31/16 13:20	Region_ID	ID	26
Stoptime	Date and time	3/31/16 13:38	Bus count	Count	16
Bikeid	ID	4622	Number of reads	Count	217
Tripduration	Seconds	1092	Speed	Miles per hour	36.82
From_station_id	ID	456			
From_station_name	Address	2112 W Peterson Ave			
To_station_id	ID	326			
To_station_name	Address	Clark St and Leland Ave			
Usertype	Subscriber/customer	Subscriber			
Gender	Male/female/blank	Male			
Birthyear	Year	1984			
Station ID	ID	456			
Station address	Address	2112 W Peterson Ave			
Station location	Latitude/longitude	41.991178/− 87.683593			
Bike capacity	Integer	15			
Online date	Date	5/12/15			

terrain map. The z coordinate values define the shape of the terrain, which creates the various curvatures throughout the terrain map (Fig. 4).

The steepness of the curvature of hill and crater is defined by the z coordinate values (hill for positive z coordinate values and crater for negative z coordinate values). The z coordinate value is determined by the number of bikes stored in each station. For example, when the target station capacity is 70%, if bike storage is greater than the target station capacity, it is considered as surplus; if bike storage is less than the target station capacity, it is considered as shortage. The z coordinate values are then calculated through the summation of the number of bikes in the grid, and the formula is as follows in Eq. (1):

$$\sum_{i=1}^{N} \begin{cases} 0 & (D_i \geq R) \\ \left(1^2 - \left(\frac{D_i}{R}\right)^2\right) * B_i & (D_i < R) \end{cases}. \tag{1}$$

The equation is used to obtain the z coordinate value of each intersection of the summation grid using the distance, D, in N stations, and the bike surplus amount B from each station i. The D_i value is calculated by subtracting x and y coordinates of the intersection of summation grids to x and y coordinates of station i. A station with a low D value will have a stronger influence in calculating the z coordinate. If the D is greater or equal to the summation range, R, the station is excluded when calculating the z coordinate.

Once the 3D terrain map of the bike usage data is created, agent-based simulation can be performed. The agent-based simulation in the proposed algorithm utilizes the curvatures of the 3D terrain map to determine the moving direction. Figure 5 illustrates how an agent rebalances the bikes from one station to another. For example, the agent (truck) illustrated as blue ball is placed in the hill (a station with congested bikes) loads

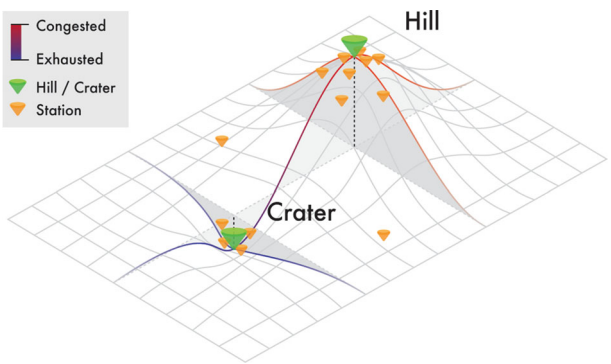

Fig. 4 Visualization of hill and crater corresponding to z coordinate: Red indicates positive z coordinate value, and blue indicates negative z coordinate value

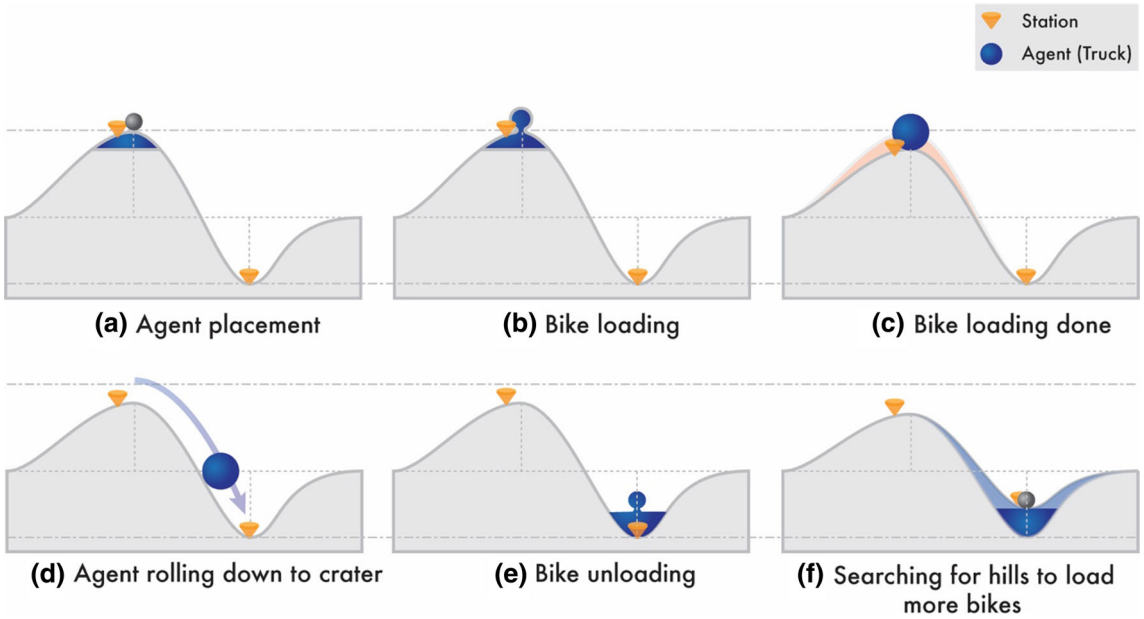

Fig. 5 Visualization of real-time truck routing

up bikes and then rolls down toward the crater (Fig. 5a–d). Then, the hill gets shallower as much as the bikes loaded onto the ball (Fig. 5e, f). When the ball reaches to the crater (a station with exhausted bikes), the ball unloads bikes. As a result, the crater is filled up. Thus, the proposed method uses the 3D terrain map to generate routes for the rebalancing truck simulation. The historical traffic congestion data are implemented on the terrain map for agent-based simulation. The congestion data were used to reflect the traffic volume of the city throughout the time and day. Because the agents in the algorithm are the trucks, it is crucial to apply actual traffic status when evaluating efficient station locations for bike redistribution. Thus, the congestion data are used to define the speed of the agent, which is the truck.

Another aspect of 3D terrain map is that it can visualize general search levels for bike storage in regional stations as well as elementary search levels. Because the terrain map is created based on bike storage status in regions adjusted by the summation range, the bike storage rate of regional stations can be easily visualized. Figure 6 illustrates the mechanism of regional bike storage rates visualization. The congested

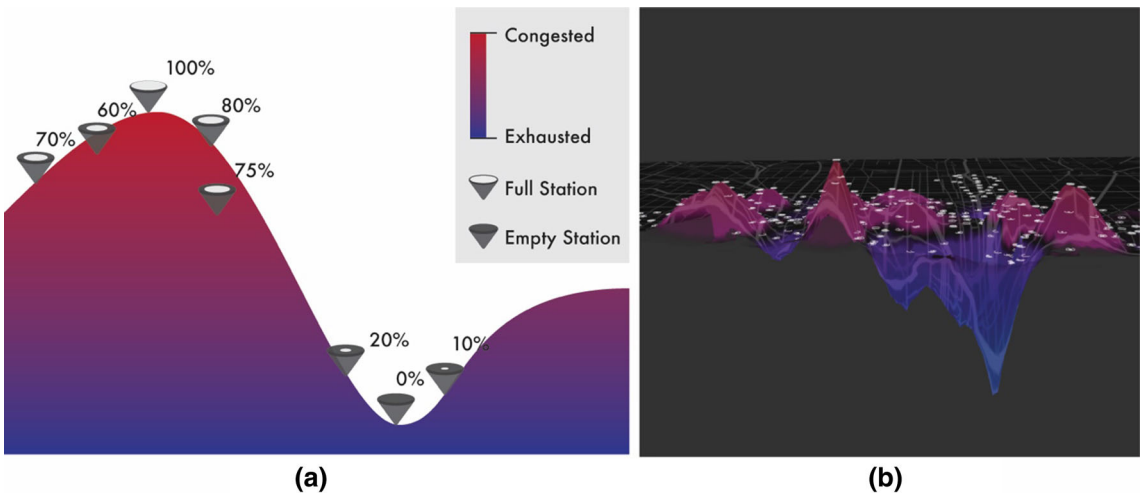

Fig. 6 **a** Visualization method for general search for bike storage in regional stations and **b** actual implementation on 3D terrain map

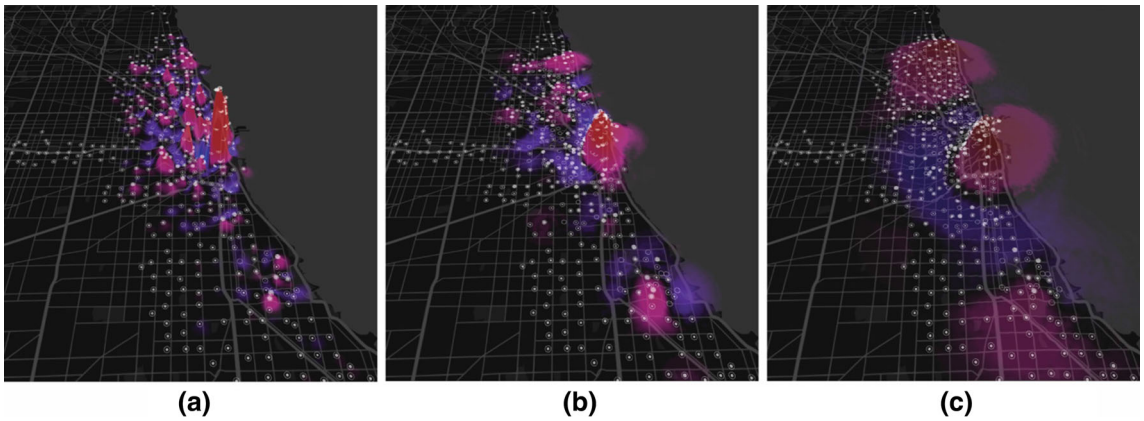

Fig. 7 **a** Small summation range; **b** medium summation range; and **c** large summation range

regions are depicted in red and exhausted regions in blue. In addition, the bike storage rate of the individual stations is visualized as the fill rate of the circle for the elementary search level.

Figure 7 illustrates the implementation of general search visualization on a 3D terrain map. As shown in the figure, the grid distance and summation range of the stations are adjustable to create gradual terrain curves. In this respect, it is possible to calculate and visualize regional imbalances in bike usage patterns by adjusting the cluster size of neighboring stations.

3.3 Curvature-based distribution algorithm

The proposed algorithm has three adjustable steps: (1) unweighted, (2) territorially weighted, and (3) periodically weighted curvature-based distribution algorithms (Fig. 8). Adjustable steps are needed because the BSSs in various cities need different strategies when operating the rebalancing trucks. For example, Tareungi, the BSS in Seoul, Korea, dynamically operates rebalancing trucks in specific city districts, because each district has its own jurisdiction. In addition, the city of Seoul operates 36 trucks with 100 operators for bike redistribution over a 24-h period with shifts. The efficiency of moving rebalancing trucks the entire day is unclear and requires heavy budgets. Conversely, Divvy, the BSS in Chicago, utilized five to seven rebalancing trucks for approximately 2400 bikes and 300 bike stations in 2014 (de Chardon et al. 2016); currently, Divvy has approximately 5800 bikes and 580 bike stations. However, some cities use static rebalancing trucks that operate overnight to reposition the moved bikes. Thus, by having the adjustable steps in the algorithm, it is possible to find an optimized truck operation strategy suitable to a specific city. The proposed algorithm has three adjustable steps: (1) unweighted; (2) territorially weighted, and (3) periodically weighted curvature-based distribution algorithms. Adjustable steps are needed because the BSS in various cities needs different strategies when operating the rebalancing trucks. For example, Tareungi, the BSS in Seoul, Korea, dynamically operates rebalancing trucks in specific city districts, because each district has its own jurisdiction. In addition, the city of Seoul operates 36 trucks with 100 operators for bike redistribution over a 24-h period with shifts. The efficiency of moving rebalancing trucks the entire day is unclear and requires heavy budgets. However, some cities use static rebalancing trucks that operate overnight to reposition the moved bikes. Thus, by having the adjustable steps in the algorithm, it is possible to find an optimized truck operation strategy suitable to a specific city.

The detailed descriptions on the algorithm are as follows:

Step 1 Stationed bike evaluation
Bike usage data contain an incoming and outgoing number of bikes per minute. The algorithm analyzes the increased and decreased number of stationed bikes per station every minute.
Step 2 Terrain map generation
Create a 3D terrain map based on the increased and decreased number of stationed bikes in each station. A higher number of surplus bikes will result in higher values of the z coordinate, resulting in higher terrain representations on the map. If additional trucks are needed, the algorithm will move to Step 3; otherwise, the algorithm will continue to Step 4.

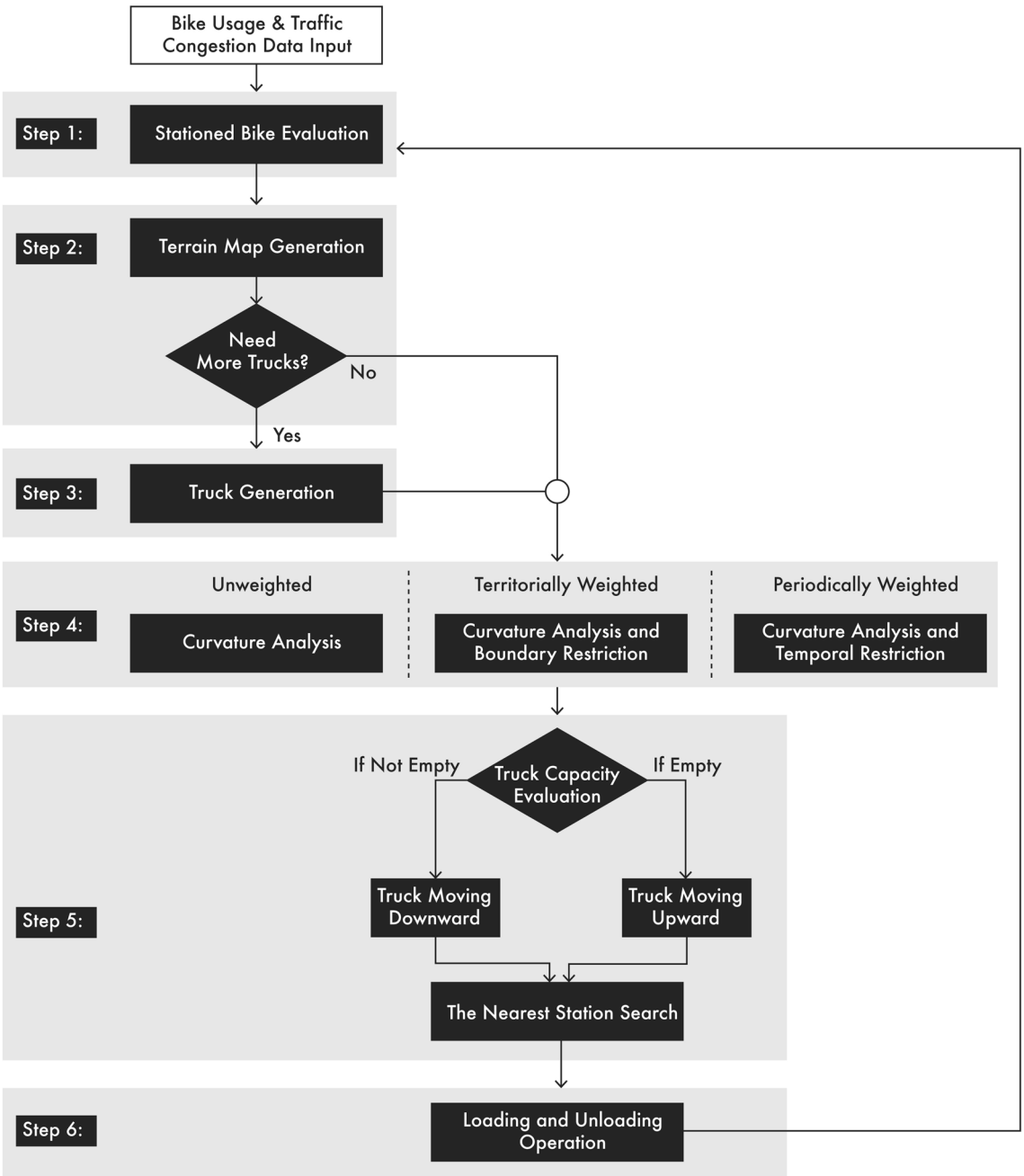

Fig. 8 Curvature-based distribution algorithm

Step 3 Truck generation
The redistribution truck is generated, placed at the station closest to the highest z coordinate, and loaded with the surplus bikes from the station. The surplus bikes are the number of extra bikes greater than 70% of the bike capacity of each station.
Step 4 Curvature analysis
The curvature defines the direction of the truck movement. Each truck calculates the gradient vector of the surface of the terrain map at its x and y positions to determine how much of the current curvature is tilted in which direction (Fig. 3c). The algorithm contains three adjustable weighting types: unweighted, territorially weighted, and periodically weighted algorithms. In the case of the territorially weighted algorithm, Steps 4 and 5 are changed. The direction of the truck movement is defined by the curvature,

and the truck searches the nearest stations in the moving direction; however, the truck only moves within the designated district of the city. Unlike both unweighted and territorially weighted algorithms, the periodically weighted algorithm operates for a designated duration. Steps 4 and 5 only operate during limited time periods. For example, for efficiency, the truck only operates from 6:00 to 24:00 because of the low BSS usage rate after 24:00.

Step 5 Truck capacity evaluation

If the truck is empty, the truck moves upward to the highest z coordinate to load bikes from the station with a high surplus of bikes. When the truck is loaded with bikes, it moves in a downward direction of the curvature gradient and searches the nearest stations. When the truck is not loaded with bikes, the truck moves upward to pick up remaining bikes to load.

Step 6 Loading and unloading operation

If there are surplus bikes at the station, more bikes are loaded to the truck with consideration of truck capacity. If there is a shortage of bikes, the bikes are unloaded from the truck. The process is repeated from Step 1 until the algorithm runs for a given time.

Agent-based simulation was conducted to find an optimal rebalancing truck operation strategy. Figure 9 illustrates the proposed algorithm in use. After the rebalancing truck is placed on the hill of the 3D terrain map, the truck loads bikes from the nearest station and rolls down in the gradient direction; next, it unloads the bikes at the stations where they are needed. If the truck has no bikes left to unload, it climbs back up to the hill and loads bikes. As a result, the hill will become shallower and the crater will be filled; hence, both will become flat.

We recorded the trucks' movement data during the agent-based simulation. The truck ID, bike loading action (target station ID, number of loaded and unloaded bikes), and time data are all saved for analysis, including bike usage patterns and optimal truck routing. Thereafter, we compared the three rebalancing strategies (unweighted, territorially weighted, and periodically weighted) to identify the most efficient and effective truck rebalancing strategy.

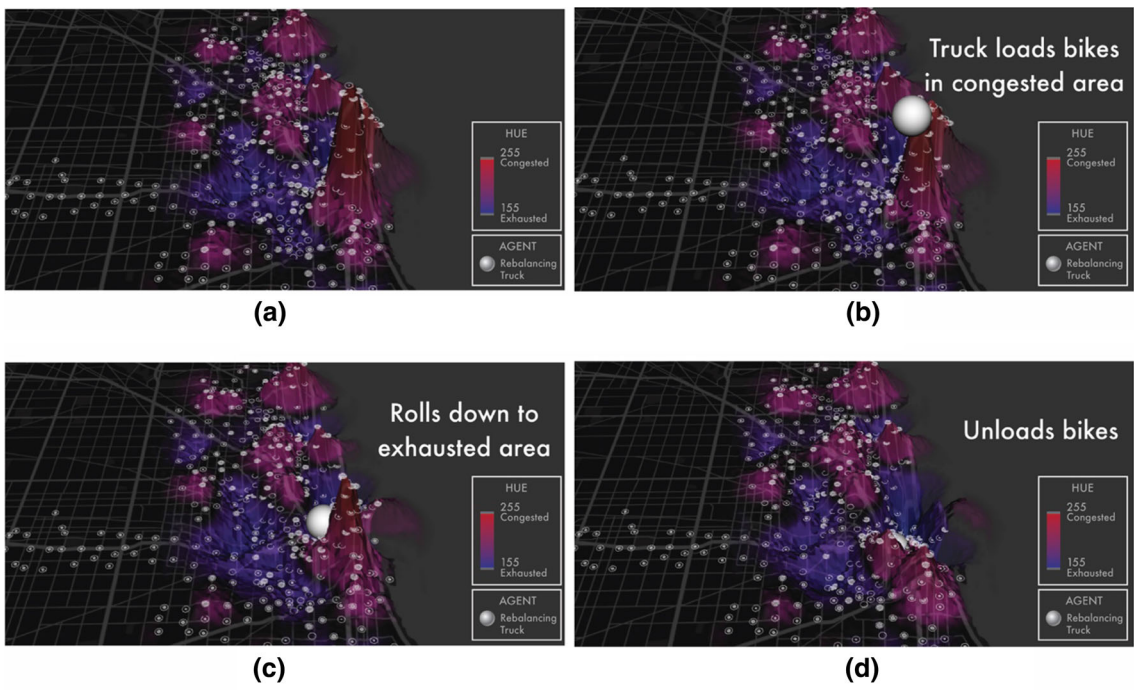

Fig. 9 Agent-based simulation example (ball size represents the number of loaded bikes)

3.4 Evaluation metrics

To measure the performance of the algorithm, we created two evaluation metrics: (1) rebalance imbalance metrics (RIM) and (2) rebalance rate (RR). The RIM calculates the distribution number of insufficient stations and the number of surplus stations; the formula is as follows:

$$\text{Rebalance Imbalance Metrics (RIM)} = \sqrt{\frac{\sum_{i=1}^{N}(B_i)^2}{N-1}} \tag{2}$$

$$B_i = C_i - T_i \tag{3}$$

The RIM is measured by calculating the distribution of bike availability rates, B, in each station i per minute. The bike availability rate is computed by comparing the difference between C, the stored bike storage rate in stations, and T, the target bike storage rate. Thus, the higher metrics mean there are many stations with an imbalanced bike storage rate, and the smaller metrics mean that bikes are well rebalanced throughout the stations. In addition, we also analyzed the rebalance rate, RR, which is RIM over time, T, to define the impact on imbalance by time of day. $\Delta\text{RIM}/\Delta T$ makes it possible to analyze how well bicycle balance is achieved over time. The formula is as follows:

$$\text{Rebalance Rate (RR)} = \frac{\Delta\text{RIM}}{\Delta T} \tag{4}$$

In addition to the proposed evaluation metrics, we also utilized three widely used evaluation metrics for bike rebalancing: (1) service failure rate; (2) number of daily rebalancing operation; and (3) daily distance covered by the trucks (Chiariotti et al. 2018). Service failure rate measures the fraction of time the system is out of service. The number of daily rebalancing operation measures the number of trips that rebalancing trucks made per day. The daily distance covered by the truck measures distance per rebalancing trips. The proposed algorithm and evaluation metrics provide three significant insights for the service operators. First, the operator can evaluate and determine how trucks should move to rebalance bike throughout the city. Second, operators can simulate the rebalancing strategy that is suitable for the city by adjusting number of trucks, truck capacity, and operating behavior. Lastly, the service operators can estimate truck operation cost based on the number of trips and daily truck distance in relation to the service availability.

4 Implementation

The curvature-based distribution algorithm proposed in this research can provide insights on bike usage pattern and rebalancing truck operation strategy through visual analysis and agent-based simulation. We used Divvy BSS datasets from January 2016 to December 2016 and historical traffic congestion data from January 2016 to December 2016. Divvy began operating in the city of Chicago in the USA on June 28, 2013, and in 2016, it operated 5748 bikes at 581 stations. The Divvy dataset consists of 3,595,383 observations, and the historical traffic congestion data consist of 1,499,126 observations. The Chicago BSS datasets were collected from the official Divvy data portal (Divvy Data Set) and the Chicago traffic congestion datasets from the official Chicago data portal (Chicago Traffic Tracker Data Set). Thus, the proposed algorithm allows the operators to simulate the truck placement and assign the moving route through the curvature-based distribution algorithm. In addition, the algorithm can test the most effective truck rebalancing strategies for Chicago.

4.1 3D terrain map optimization

To apply the proposed curvature-based distribution algorithm in this study, a 3D terrain map generation is necessary to reflect the station bike capacity status for every time sequence. The Divvy dataset contains latitude and longitude location information for each station and also has information on where to rent the bike, where to return it for each trip, and each time frame. We transformed the dataset as a 3D terrain model to visualize changes in the bike holding capacity of every station minute to minute. The data from January 2016 to December 2016 contain a total of 525,600 (365 days * 24 h * 60 min) terrain models. The z coordinate of each intersection is determined based on Eq. (1) in a two-dimensional grid representing the terrain as described in Sect. 3.2.

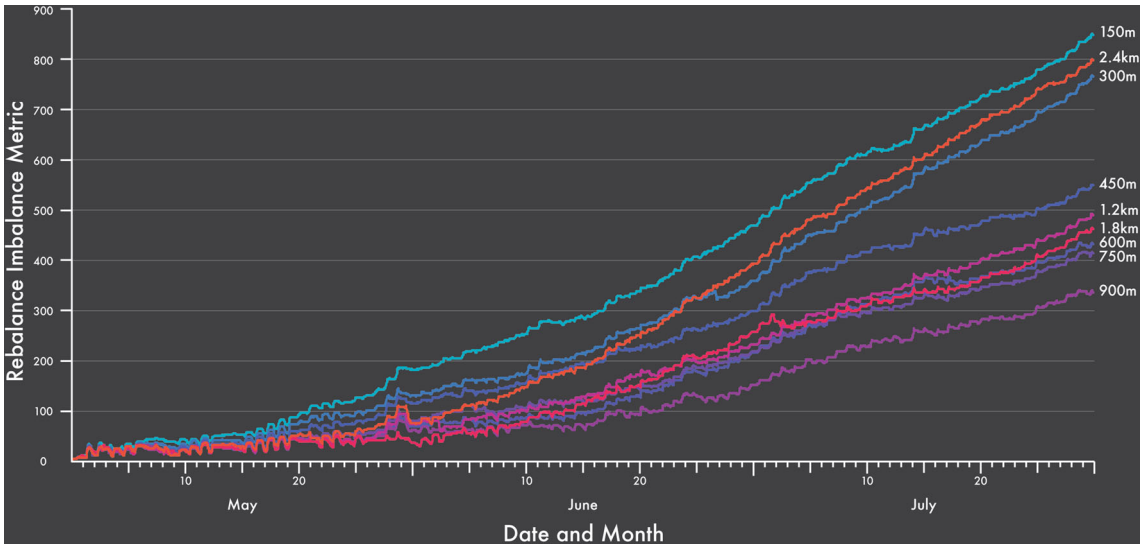

Fig. 10 Rebalance imbalance metric with various summation distances

In this method, the shape of the terrain model is influenced by the degree of summation range, R. Because the change in the summation range changes the overall curvature shape, we proceeded to find the summation range that performs the proposed distribution algorithm most effectively. Based on the distribution algorithm, 30 trucks were simulated, and their bike balance metrics were measured by changing the summation range to 150 m, 300 m, 450 m, 600 m, 750 m, 900 m, 1.2 km, 1.8 km, and 2.4 km. As shown in Fig. 10, the performance of the distribution algorithm shows the highest efficiency at the appropriate summation range, rather than when the range is too low or too high. The low summation range allows the truck to accurately detect the congested status of the surrounding stations and allow detailed and efficient truck movement. However, it does not allow detecting bike congestion in terms of neighboring stations. The high summation range allows us to identify the overall tendency and resolve regional disparities; however, it is less effective in detecting and resolving imbalances in individual stations. The simulation results show that the summation range of 900 m is most suitable for BSS in Chicago.

4.2 Visual analysis on bike usage pattern

The results of the curvature-based distribution algorithm can be dynamically visualized to evaluate bike usage patterns. The advantage of dynamic visualization is that it can incorporate spatiotemporal information to analyze data in sequence. Considering this factor, bike usage patterns per time and day of the week are analyzed. The proposed algorithm utilizes the 3D terrain map in two major ways. First, the 3D terrain map is used to identify truck routing direction for agent-based simulation. Second, the map is used to calculate a heatmap of the general search level of the bike storage rate in neighboring stations over time. Thus, the 3D terrain map is used for simulation and calculation; however, only the top view of the terrain map is used for the visual analysis of the bike usage pattern. Because the proposed algorithm can evaluate both elementary and general search levels for station storage rates, we visualized both the bike storage rate of individual stations and the dynamics of bike storage rates in neighboring stations. Figure 11 shows a sample visualization of Chicago in July. To visualize the elementary component, we represented the percentage of bike storage status as the size of the white circle in the map (Urbica 2016). To visualize the bike storage rate of regional stations, we generated a heat map based on the curvature height as explained in Sect. 3.2. We created a visualization map with identical length and width that includes all the locations of the bike stations that existed in Chicago in 2016. As shown in Fig. 11, the visualization map was created with latitude − 87.8959 to − 87.4563, longitude 42.0635 to 41.7569. The color of the heat map was visualized to be 155 (exhausted), 205 (target), 255 (congested) on the basis of the hue value 155–255 on the hue, saturation, and brightness (HSB) scale.

The visual analysis was conducted by averaging bike usage per day for summer and winter. The results of the visual analysis show different usage patterns in both time and day of the week, varying in different

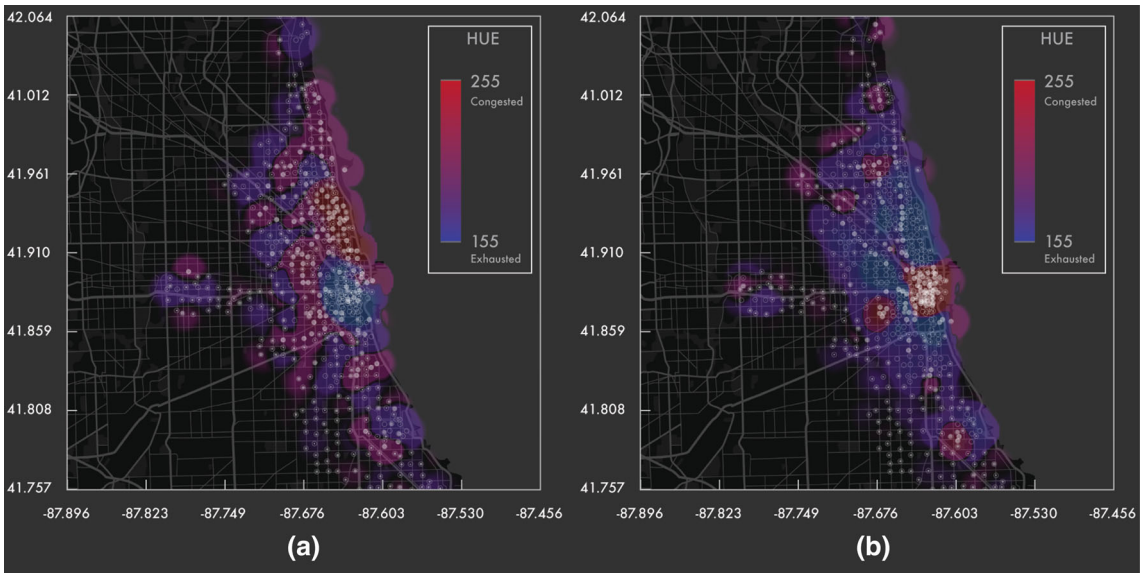

Fig. 11 Visual analysis sample of Chicago's BSS usage

seasons, winter and summer (Figs. 12, 13, respectively). Both visualizations illustrate stochastic bike usage patterns at three different hours, 9:00, 14:00, and 20:00. The color range (blue–purple–red) represents bike storage rate in regions. The blue indicates an exhausted area, and red indicates a congested area. The teal graph on the bottom of the figures represents the number of bike trips in a day. The orange graph is the rebalance rate, RR. Thus, the lower values in the orange graph indicate a more stable bike return-to-rent ratio value. In each season, the users of Divvy, the BSS in Chicago, tend to ride their bikes more on weekdays compared to weekends. The visual analysis results show that Divvy users actively use the BSS for commuting purposes. At approximately 08:00, the bike renting rate is high around the downtown area, and the return rate is high in the downtown where offices are concentrated. For example, a significant commuting tendency is visualized from Humboldt–Garfield Park E/W to Rogers Park West Ridge. At 18:00, the bike renting rate is high in the downtown area. The Divvy users may use the BSS for commuting within the station range and use public metro for long distance travel from the outer city. However, unlike on weekdays, the overall usage of the service is significantly decreased on the weekends. As shown in Figs. 12 and 13, bike usage in the winter is especially low (total 30,531 bike trips), compared to the summer (total 110,908 bike trips). Bike usage in the winter is 27.5% of the bike usage in summer. In the summer, bike riders tend to use Divvy on the weekend to visit the beach area. Despite the high usage rate during the weekend, the return-to-rent ratio value of the bike is well balanced. This may result because most weekend entertainment facilities are located in the downtown and beach areas. Divvy users rent bikes in the downtown area and travel back and forth between downtown and the beach.

4.3 Agent-based simulation for rebalancing bike sharing system

The rebalancing trucks help to redistribute bikes in every station throughout the city. However, it is difficult for the service operator to manage trucks to efficiently and effectively load the bikes from a station and unload them where they are needed. In addition, the most effective truck allocation duration and distribution strategies are not apparent. Seoul city, for example, operates 36 trucks for 24 h with approximately 100 workers. Furthermore, each district operates its own trucks; therefore, it is difficult to move the bikes from one district to another. We conducted a simulation using three weight settings (unweighted, territorially weighted, and periodically weighted algorithms). All of the simulations share variable settings, including the number of trucks (10, 20, 30, and 40 trucks), truck speed (weighted based on the traffic congestion data), and truck capacity (25 bikes per truck). The periodically weighted strategy in this simulation operated the trucks for 18 h, 6:00 to 24:00, identical to Divvy's rebalancing operation hours (de Chardon et al. 2016). The current BSS is managed by a system operator who directs truck drivers to move the bikes from the congested stations to exhausted stations. The ideal target for bike distribution throughout the city is to maintain the

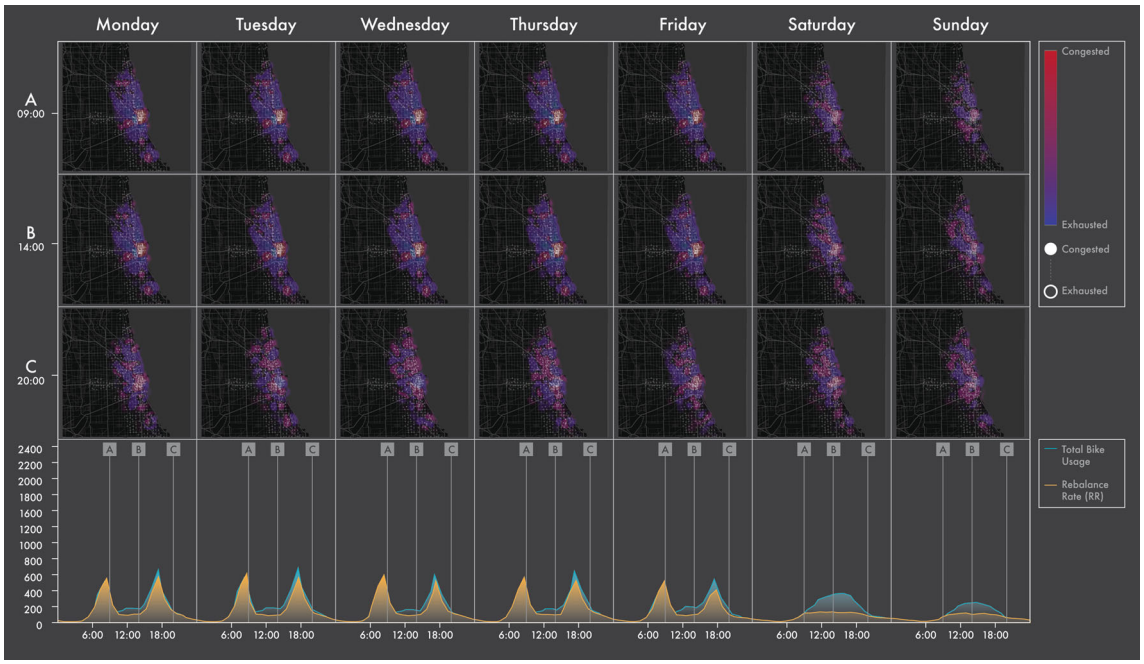

Fig. 12 Visualization of bike usage patterns in January

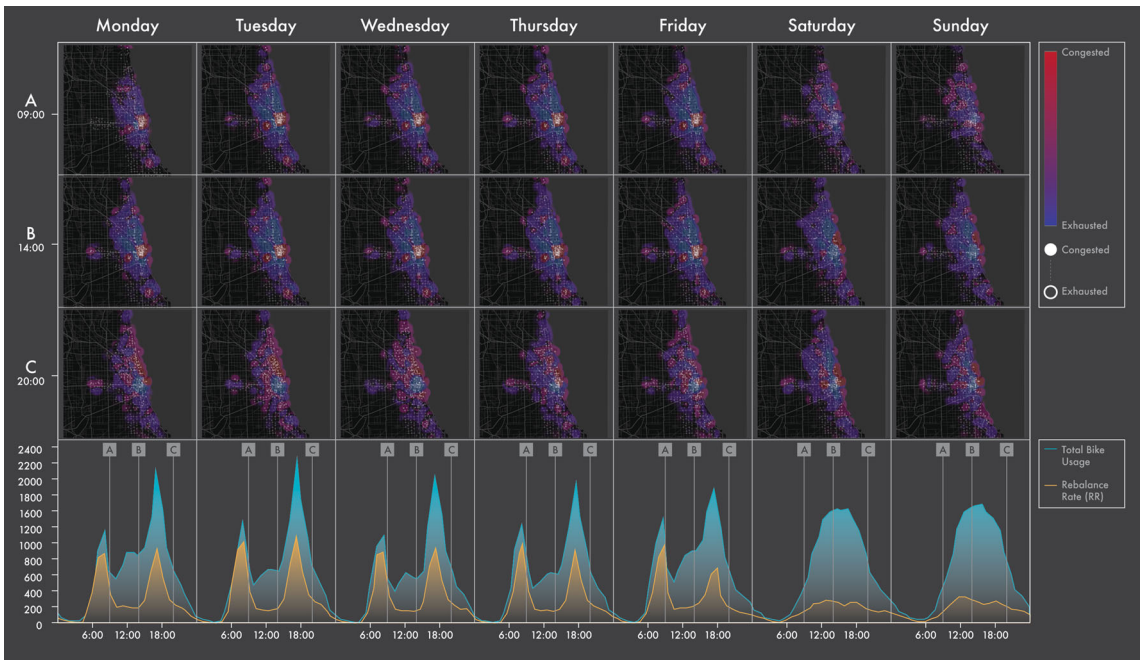

Fig. 13 Visualization of bike usage patterns in July

same station capacity level; therefore, the increase and decrease in the number of bikes per station are calculated every minute.

We tested the performance of three bike rebalancing strategies and four variations in truck numbers based on how well Chicago's Divvy bikes were balanced. To see how well bike storage in each station is maintained at the target level, we compared and analyzed the RIM and service failure rate described in Sect. 3.4. To test the efficiency of the rebalancing trucks, we compared the result of the simulation with reference data that were simulated without operating trucks (Table 3). In addition, we considered the

Table 3 RIM and service failure rate evaluation results of different truck operation strategies

Evaluation metric	Strategy	Truck	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
Rebalance imbalance metric	Ref.	0	147.51	217.50	250.87	319.68	434.55	639.51	746.78	580.58	671.29	409.57	312.27	149.67
	Unweighted	10	21.54	31.22	102.78	66.22	220.83	419.75	350.53	369.91	332.26	241.98	77.10	32.89
		20	16.70	21.38	34.75	36.55	60.16	164.35	263.88	160.13	169.94	114.74	42.70	23.91
		30	16.23	17.79	21.87	27.95	46.33	136.32	70.08	101.89	92.65	61.19	28.23	21.84
		40	9.82	14.96	18.20	20.24	37.25	53.15	55.26	61.82	51.55	34.09	23.83	18.46
	Territorially	10	35.55	45.56	129.42	178.77	307.16	526.13	638.73	473.64	535.02	284.94	124.89	34.23
		20	23.17	24.49	53.80	73.55	187.06	400.76	399.60	429.42	429.52	241.24	90.88	28.42
		30	14.55	16.56	36.50	36.06	65.30	142.54	89.15	187.66	124.10	91.91	38.14	23.99
		40	14.70	17.49	37.03	39.95	69.28	235.71	169.79	173.79	122.00	80.51	42.63	22.70
	Periodically	10	22.8.3	28.71	63.37	93.86	116.58	293.04	434.73	360.95	337.33	262.90	74.48	30.30
		20	15.27	25.72	32.07	35.41	61.49	183.70	188.62	144.11	123.60	176.82	41.81	23.86
		30	12.28	14.11	23.12	27.27	42.69	85.16	70.66	87.77	87.61	58.67	33.14	19.60
		40	13.52	11.22	17.28	20.70	34.09	60.01	50.54	79.24	55.51	34.50	25.84	20.22
Service failure rate	Ref.	0	0.34	0.39	0.42	0.43	0.50	0.54	0.56	0.58	0.56	0.53	0.52	0.43
	Unweighted	10	0.04	0.06	0.23	0.16	0.27	0.34	0.34	0.41	0.32	0.35	0.20	0.07
		20	0.02	0.03	0.08	0.09	0.14	0.23	0.28	0.27	0.26	0.22	0.11	0.03
		30	0.02	0.02	0.03	0.05	0.11	0.19	0.15	0.20	0.16	0.12	0.05	0.03
		40	0.01	0.02	0.02	0.03	0.07	0.09	0.12	0.14	0.11	0.07	0.04	0.02
	Territorially	10	0.06	0.11	0.22	0.23	0.31	0.40	0.41	0.48	0.38	0.3.3	0.23	0.08
		20	0.04	0.04	0.11	0.13	0.24	0.33	0.30	0.36	0.30	0.31	0.19	0.05
		30	0.02	0.02	0.08	0.07	0.14	0.21	0.19	0.26	0.23	0.18	0.08	0.03
		40	0.02	0.02	0.08	0.07	0.12	0.22	0.19	0.24	0.17	0.14	0.09	0.03
	Periodically	10	0.04	0.06	0.16	0.18	0.21	0.29	0.32	0.38	0.32	0.38	0.22	0.05
		20	0.02	0.04	0.07	0.09	0.14	0.24	0.24	0.27	0.22	0.30	0.11	0.03
		30	0.01	0.02	0.04	0.05	0.10	0.15	0.16	0.18	0.16	0.12	0.07	0.02
		40	0.01	0.01	0.02	0.03	0.07	0.11	0.11	0.16	0.12	0.07	0.04	0.02

Table 4 Number of daily operation and daily trip distance evaluation results of different truck operation strategies

Evaluation metric	Strategy	Truck	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
Number of daily operations	Unweighted	10	99.48	106.36	51.57	102.92	60.56	53.14	73.75	52.64	87.81	59.62	87.06	73.25
		20	88.70	109.20	93.97	115.16	119.76	92.46	63.78	84.08	75.64	75.68	105.29	81.06
		30	79.80	93.18	112.66	107.52	107.73	82.12	119.87	106.65	112.48	117.84	113.97	73.02
		40	72.59	91.75	106.20	111.47	111.49	145.74	123.96	119.20	139.28	135.30	115.11	66.13
	Territorially	10	64.94	72.29	51.16	57.96	50.34	46.71	45.87	33.97	54.59	61.47	68.99	65.06
		20	58.55	70.86	64.87	65.37	49.53	44.47	62.07	50.13	58.61	45.02	55.51	55.90
		30	67.68	78.76	73.53	80.66	84.12	79.19	103.58	70.51	89.07	81.24	100.34	65.70
		40	54.85	60.75	57.52	69.20	74.28	57.88	75.42	64.57	86.25	80.81	74.76	51.23
	Periodically	10	92.06	95.91	91.35	85.95	103.63	82.88	79.46	63.35	87.29	49.95	83.88	83.34
		20	89.37	100.92	102.63	112.38	128.52	81.80	78.41	86.21	109.24	52.96	111.80	75.31
		30	80.85	95.73	113.20	115.12	126.16	125.80	126.17	113.36	114.21	110.99	115.76	70.93
		40	70.70	89.24	107.99	109.41	124.04	135.30	135.98	101.15	123.71	135.86	112.39	66.58
Daily trip distance (km)	Unweighted	10	328.90	297.59	475.47	410.03	392.20	415.10	405.28	385.22	364.30	551.89	322.75	277.44
		20	318.46	350.87	274.36	369.00	329.95	366.40	378.19	369.59	364.59	360.73	323.64	284.40
		30	294.49	294.31	336.23	324.03	285.00	317.37	260.28	288.81	262.31	304.36	293.36	293.09
		40	321.48	295.38	315.09	305.55	304.41	294.45	246.22	276.93	293.82	297.76	306.91	299.00
	Territorially	10	203.44	197.18	254.03	261.36	256.68	190.29	128.80	269.22	180.00	202.38	298.32	202.36
		20	201.28	195.55	190.79	193.82	235.62	229.54	179.72	192.12	177.90	277.23	206.88	237.69
		30	247.07	231.89	229.11	234.50	229.89	224.86	206.57	260.03	210.73	244.48	266.39	266.89
		40	178.70	167.12	193.58	206.54	165.18	219.73	168.10	190.34	190.68	180.28	208.60	201.53
	Periodically	10	217.58	200.25	244.84	304.82	219.63	291.82	148.54	304.66	243.32	417.81	214.06	205.23
		20	214.37	231.49	205.78	264.22	225.76	257.09	245.34	248.90	191.81	349.38	237.95	203.68
		30	215.34	197.77	221.19	226.94	223.13	190.53	179.53	199.68	185.74	219.87	220.76	180.50
		40	220.15	199.94	207.24	225.93	202.73	197.31	177.66	189.36	183.19	204.80	216.69	205.26

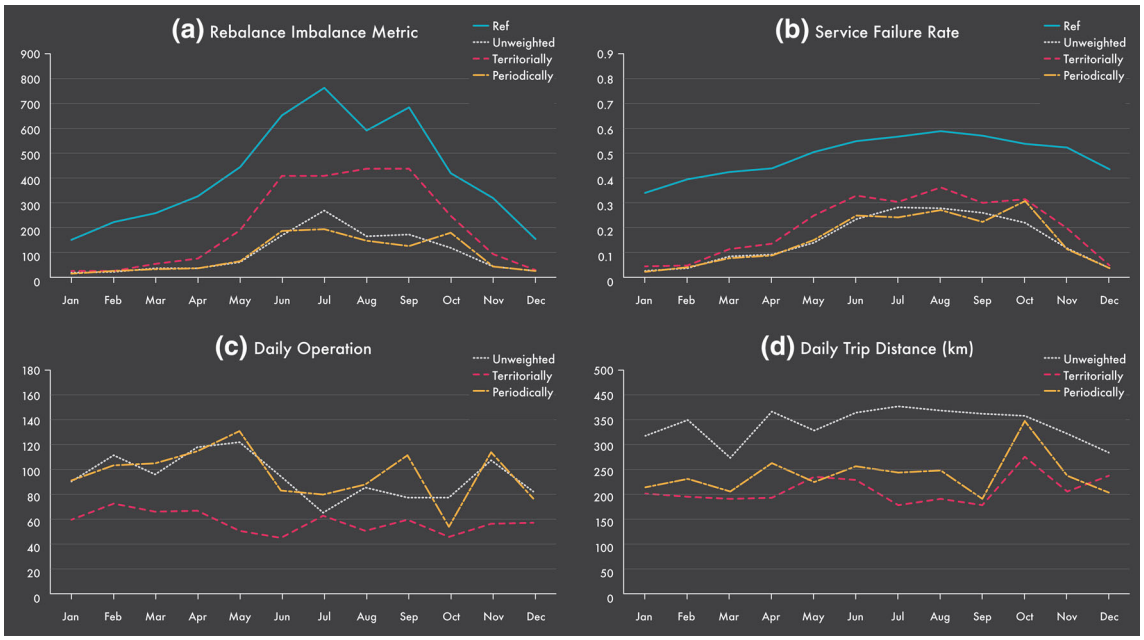

Fig. 14 Evaluation results of three rebalancing strategies over 1 year

number of daily operations (daily number of loading or unloading bikes by one truck) and the average daily travel distance per truck to observe the detailed behavior of the truck (Table 4).

The first noticeable tendency in RIM results is seasonal changes. As mentioned in Sect. 4.2, bicycle usage varies greatly according to the season. It is more difficult to rebalance in summer because the number of bicycle trips is greater in summer than in winter. RIM values are also generally higher in summer than in winter. As illustrated in Fig. 14a, we graphed the change in RIM values of 20 trucks over 12 months. In the winter, all rebalancing strategies show a low RIM value of 10–15% compared to the reference. In the summer with increased bike usage, it can be seen that the territorially weighted strategy shows about 50% RIM compared to the reference. The RIM value of unweighted and periodically weighted strategies is relatively lower at 25%. Second, the tendency, according to the number of trucks, indicates a lower RIM value as the overall number of trucks increases. The largest change usually occurs when increasing the number of trucks from 10 to 20. RIM decreases even when the number of trucks is increased to 30 and 40; however, its influence tends to decrease gradually. Regarding the aspect of trucks and seasonal changes mentioned above, the territorially weighted strategy in summer appears to be 20%; unweighted and periodically weighted strategies have a low RIM level of 6–7%. This shows that unweighted and periodically weighted strategies outperform the territorially weighted strategy even if the number of trucks changes. The outstanding performance of the unweighted strategy can be explained by its characteristics of utilizing the truck to rebalance bikes from various districts in Chicago. The truck globally seeks for congested stations and unloads it to the station where in need. However, unlike the unweighted strategy, the trucks with territorially weighted strategy only rebalance bikes within designated districts. The territorially weighted strategy is one of the most common truck operation strategies owing to its efficiency where the driver can operate following the same routine. However, operation radius limitations interfere with the ability of the trucks to transport bikes in different districts; therefore, there is an increase in imbalanced bike stations. In real-world operations, if the truck is operated across the districts, detailed and intuitive operator's instructions must be provided to distribute the trucks smoothly. Thus, it is necessary to investigate how often these detailed operator instructions can be reliably processed through the simulation in a short period of time.

The role of the rebalancing truck is to maintain the distribution of bikes throughout the city; however, it is also to provide BSS consistent and continuous to the users. When a user visits a station to rent a bicycle and the bicycle is not available, the users lose service availability which greatly affects the quality of service. We used the bicycle rent and return data to calculate the service failure rate by measuring whether the user had the available bike at the station. If a bike is not available at the time of the action, we classified

the state as an event and counted the number of times when the station had no bikes left. The results of the service failure rate show seasonal differences as the result of RIM. As shown in Fig. 14b, unweighted, territorially weighted, and periodically weighted strategies perform at a low service failure rate (Ref = 0.34; Unweighted = 0.02; Territorially weighted = 0.04; Periodically weighted = 0.02) in January despite the cold weather, with low BSS usage. However, the service failure rate significantly increased during July (Ref = 0.56; Unweighted = 0.29; Territorially weighted = 0.30; Periodically weighted = 0.24), when BSS usage is also significantly increased. An interesting result is that the territorially weighted strategy has a lower performance on service failure than the other two strategies, similar to the RIM result; however, the difference is not as large as the RIM. This implies that the imbalance condition of individual stations is more severe than the insufficient state of the stations in the neighboring stations. One of the characteristics of the territorially weighted strategy is that it rebalances only within the designated districts. As the number of imbalanced stations increases, the number of trucks should be concentrated in a district immediately, which can make the operation inefficient.

We also analyzed the number of operations and the trip distance of the truck to evaluate the behavior pattern of the proposed agent-based simulation. Truck operation and trip distance affect operational difficulty and maintenance costs. If the number of operations in one day is too large, the truck drivers' workload will increase, and if the trip distance is too long, the truck maintenance including fuel expenses will increase. We input three minutes as the time it takes to perform an operational action. Unlike RIM and service failure rate, the truck behavior pattern is significantly influenced by the seasons (Fig. 14c, d). It is important to note that unweighted and periodically weighted strategies, which presented similar performances at RIM and service failure rate, demonstrated a significant difference in trip distance. In other words, operating trucks during the appropriate hours can significantly influence the trip distance; therefore, operating trucks for 24 h is not an efficient strategy. Considering these factors, the truck can be operated more efficiently with less time and less travel distance.

5 Conclusion

This research proposes a novel algorithm to rebalance BSS. Unlike the literature on the rebalancing problem, our method utilizes agent-based simulation with a 3D terrain map. The 3D terrain map was created based on bike usage data and traffic congestion data. The agent-based simulation is conducted by utilizing curvatures on the 3D terrain map as a trajectory for rebalancing truck routes. The proposed algorithm can simulate multiple agents with large volumes of data compared to the literature on bike rebalancing. Importantly, the algorithm allows the simulation of various truck operation strategies with different numbers of trucks, truck working ranges, and truck working hours to find suitable strategies for different cities. We simulated the proposed algorithm with Chicago's bike trip data to find a suitable rebalancing strategy for the city. We simulated three rebalancing strategies (unweighted, territorially weighted, and periodically weighted) with four different number of rebalancing trucks (10, 20, 30, and 40) to find an optimum rebalancing strategy for Chicago. Despite the seasonal differences, the algorithm successfully rebalanced bikes throughout the year. During our research, we observed two major areas worthwhile to investigate in the future. First, the real-time variation in grid summation range in 3D terrain map can help to result in more effective rebalancing the bikes. The curvature-based algorithm currently uses a fixed grid summation range to create gradual curvatures for the 3D terrain map. The curvature is created by averaging the bike accessibilities from the stations near the summation range. Therefore, the gradual curvature can help to efficiently move the bikes from a surplus station in a region to a station where they are needed, though stations in the region that have bike shortages may be disregarded. The real-time adjustment of the summation can help resolve the issues. Lastly, the proposed algorithm is capable of applying a predictive model for improvements. A predictive model can be applied when creating the 3D terrain map to find proactive rebalancing truck routes. Machine learning can be used to forecast the bike return and rent rate of stations based on bike usage, traffic congestion, weather, date, and time data. When the 3D terrain map based on the prediction is created, the agent-based simulation can help to derive optimal rebalancing truck routes.

Acknowledgements This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP, Ministry of Science, ICT and Future Planning) (NRF-2017R1C1B5018240).

References

Andrienko N, Andrienko G, Gatalsky P (2003) Exploratory spatio-temporal visualization: an analytical review. J Vis Lang Comput 14:503–541

Batty M (2001) Exploring isovist fields: space and shape in architectural and urban morphology. Environ Plan B Plan Des 28:123–150

Chemla D, Meunier F, Calvo RW (2013) Bike sharing systems: solving the static rebalancing problem. Discret Optim 10:120–146

Chiariotti F, Pielli C, Zanella A, Zorzi M (2018) A dynamic approach to rebalancing bike-sharing systems. Sensors 18:512

Chicago Traffic Tracker Data Set. https://data.cityofchicago.org/Transportation/Chicago-Traffic-Tracker-Historical-Congestion-Esti/emtn-qqdi. Accessed 3 July 2018

Conticelli E, Santangelo A, Tondelli S (2018) Innovations in cycling mobility for sustainable cities. In: Town and infrastructure planning for safety and urban quality: proceedings of the XXIII international conference on living and walking in cities (LWC 2017), 2017, Brescia, Italy, pp 155–162

de Chardon CM, Caruso G, Thomas I (2016) Bike-share rebalancing strategies, patterns, and purpose. J Transp Geogr 55:22–39

Divvy Data Set. https://www.divvybikes.com/system-data. Accessed 24 April 2018

Do M, Noh YS (2014) Analysis of the affecting factors on the bike-sharing demand focused on Daejeon City. J Korean Soc Civ Eng 34:1517–1524

Dötterl J, Bruns R, Dunkel J, Ossowski S (2017) Towards dynamic rebalancing of bike sharing systems: an event-driven agents approach. In: Portuguese conference on artificial intelligence, pp 309–320

Hillier B, Hanson J (1984) The social logic of space. Cambridge University Press, Cambridge

Hyun KH, Min A, Kim S-J, Lee J-H (2016) Investigating cultural uniqueness in theme parks through finding relationships between visual integration of visitor traffics and capacity of service facilities. Int J Arch Comput 14:247–254

Kloimüllner C, Papazek P, Hu B, Raidl GR (2014) Balancing bicycle sharing systems: an approach for the dynamic case. Eur Conf Evol Comput Comb Optim. Springer, Berlin, pp 73–84

Koblin A (2009) Flight pattern. http://www.aaronkoblin.com/project/flight-patterns/. Accessed 1 Aug 2018

Lee D, Offenhuber D, Duarte F, Biderman A, Ratti C (2018) Monitour: tracking global routes of electronic waste. Waste Manag 72:362–370

Min DA, Hyun KH, Kim S-J, Lee J-H (2017) A rule-based servicescape design support system from the design patterns of theme parks. Adv Eng Inform 32:77–91

O'Brien O (2013) Bike share map. http://bikes.oobrien.com/seoul/#zoom=15&lon=126.9935&lat=37.5621. Accessed 1 Aug 2018

O'Brien O (2017) Social benefits from public bike share data. http://oobrien.com/2017/10/social-benefits-from-public-bike-share-data/. Accessed 1 Aug 2018

Pal A, Zhang Y (2017) Free-floating bike sharing: solving real-life large-scale static rebalancing problems. Transp Res Part C Emerg Technol 80:92–116

Pan L, Cai Q, Fang Z, Tang P, Huang L (2018) Rebalancing dockless bike sharing systems. https://arxiv.org/pdf/1802.04592.pdf. Accessed 23 Nov 2018

Pei W, Wu Y, Wang S, Xiao L, Jiang H, Qayoom A (2018) BVis: urban traffic visual analysis based on bus sparse trajectories. J Vis 21:873–883

Rainer-Harbach M, Papazek P, Hu B, Raidl GR (2013) Balancing bicycle sharing systems: a variable neighborhood search approach. In: European conference on evolutionary computation in combinatorial optimization. Springer, Berlin, pp 121–132

Regue R, Recker W (2014) Proactive vehicle routing with inferred demand to solve the bikesharing rebalancing problem. Transp Res Part E Logist Transp Rev 72:192–209

Tang M (2018) From agent to avatar. In: Conference of the association for computer-aided architectural design research in Asia (CAADRIA 2018), Beijing, China. Tsinghua University, Beijing, pp 503–512

Tang M, Hu Y (2017) Pedestrian simulation in transit stations using agent-based analysis. Urban Rail Transit 3:54–60

Tufte E (2011) The visual display of quantitative information. Graphics Press, Cheshire

Turner A, Penn A (1999) Making isovists syntactic: isovist integration analysis. In: 2nd International symposium on space syntax, Brasilia

Turner A, Doxa M, O'Sullivan D, Penn A (2001) From isovists to visibility graphs: a methodology for the analysis of architectural space. Environ Plan B Plan Des 28:103–121

Urbica (2016) City bike rebalanced. https://medium.com/@Urbica.co/city-bike-rebalanced-92ac61a867c7. Accessed 23 Nov 2018

Xie C, Ma G, Li Q, Xun J, Dong J (2016) Visual exploration of tsunami evacuation planning. J Vis 19:475–487

Yan Y, Tao Y, Xu J, Ren S, Lin H (2018) Visual analytics of bike-sharing data based on tensor factorization. J Vis 21:495–509

Yuksel ME (2018) Agent-based evacuation modeling with multiple exits using neuroevolution of augmenting topologies. Adv Eng Inform 35:30–55

Zhang L, Tang S, Yang Z, Hu J, Shu Y, Cheng P, Chen J (2016) Demo: data analysis and visualization in bike-sharing systems. In: Proceedings of the 14th annual international conference on mobile systems, applications, and services companion. ACM, pp 128–128