Optimizing Station-Based Shared E-Bike Allocation Using Multi-Armed Bandit Algorithms

Wenkui Zhao

School Of Mathematics And Statistics, Wuhan University Of Technology, Wuhan, 430070, China

Abstract. In the era of rapid urban transformation, the sharing economy has revolutionized transportation solutions, with shared bicycles emerging as a prominent example. Shared bicycles have become a revolutionary mode of transportation, effectively addressing urban mobility issues by providing an affordable and convenient "last mile" travel option. Despite their growing popularity, a critical challenge remains: the optimal dynamic allocation of e-bikes, particularly in station-based sharing systems. This research tackles this complex problem by leveraging multi-armed bandit algorithms, a cutting-edge approach to intelligent resource distribution. The study conducted comprehensive experiments using real-world data from New York City's bike-sharing system in February, employing sophisticated techniques such as Upper Confidence Bound, Thompson sampling, and the innovative Sliding Window Thompson Sampling. The research meticulously evaluates algorithm performance through metrics including cumulative regret, optimal arm click rate, and Mean Absolute Error (MAE). Notably, the findings reveal the remarkable potential of multi-armed bandit algorithms in solving dynamic allocation challenges, with Sliding Window Thompson Sampling demonstrating exceptional adaptability, especially when confronting sudden environmental changes. This study provides a robust, data-driven methodology for e-bike allocation, representing a significant step towards optimizing urban mobility infrastructure and enhancing the efficiency of shared transportation systems.

1 Introduction

The sharing economy is a new economic model. It transforms how people use resources and exchange value. Bike-sharing is a key part of this model. Services like Citi Bike and OFO use mobile technology and low costs to dominate urban transport. They significantly improve urban micro-travel. Cycling e-bikes can achieve at least moderate intensity physical activity levels and contribute to physical fitness [1]. Besides, E-bikes can reduce carbon emissions and alleviate urban traffic congestion. Some studies have even found the potential benefits of E-bikes on local power systems [2]. E-bikes have evolved into a sustainable transportation option for millions worldwide, providing a low-carbon solution to the "last mile" problem [3]. However, in the actual operation of shared bicycles, how to reasonably allocate bicycles to meet the needs of different regions and at various times has always been an urgent problem

353664@whut.edu.cn

that needs to be solved. Currently, many cities adopt a fixed strategy for bicycle allocation, but it cannot effectively cope with demand fluctuations, resulting in uneven distribution of bicycles. Some areas often have a shortage of bicycles, while other regions face a waste of resources. Therefore, optimizing the allocation of shared bicycles is still a challenge.

This study aims to use the multi-armed bandit algorithm(MAB)in reinforcement learning to dynamically optimize the allocation of Citi Bike in New York City, a station-based shared bike system, thereby improving resource utilization efficiency and maximizing benefits. This research first utilizes the two baseline algorithms, Upper Confidence Bound(UCB) and Thompson Sampling(TS), to explore the stable multi-armed bandit machine and then uses Thompson's variant, Sliding Window Thompson Sampling(SW-TS), to explore the non-stationary situation, ensuring the scientific nature of the dynamic allocation in the face of sudden factors. Due to these algorithms' dynamic learning mechanism and exploration-exploitation balance, they can provide intelligent and flexible dynamic allocation solutions for urban E-bike systems.

2 Related Work

Currently, there are two mainstream E-bike operating modes: station-based and free-float. Chenyi Fu et al. compared these two models and pointed out that the station-based model has low management costs, while the free-floating model has high operating costs due to the random parking of bicycles [4]. Hao Luo et al. found that the greenhouse gas emissions of the station-based system are lower than those of the free-float system, which is more in line with the concept of sustainable development [5]. Based on the above research, the research will also adopt a station-based model. Furthermore, Mingzhuang Hua et al.classified the stations using index-based (K-means, FCM) and shape-based (DTW, K-shape) algorithms. They revealed the types of stations and their spatial distribution characteristics. Based on the distribution of urban space, they made different deployment plans for stations at various distances from the city center [6].E. Angelelli et al. proposed a simulation framework for a station-based sharing system. The framework employs a Dynamic Bicycle Rebalancing Problem (DBRP) to optimize vehicle routes, integrating user demand prediction and stochastic modeling to minimize bicycle and parking space shortages. Using the Bicimia system in Brescia, Italy, the optimal configuration of 2 morning and 1 afternoon vehicles reduces shortage time by 36%. They made detailed deployments for specific cities [7]. Sergio Jara-Díaz et al. constructed a microeconomic model of Station-Based Bike-Sharing System (SBSS), which provided theoretical support for the pricing strategy and spatial fairness of SBSS. They integrated user time and operator costs for the first time [8]. Combined with the above research, the following problems have not been solved: the first point is the adaptability to dynamic environments, the second point is the universality of cities, and the third is that it is impossible to change the solution in real time. The multi-armed bandit algorithm takes the above issues into account. The multi-armed bandit has the function of exploration and exploitation trade-off and real-time learning. One of its core algorithms, such as the Thompson algorithm, is very suitable for studying dynamic environments [9]. Multi-armed and contextual multi-armed bandits have made essential contributions to recommendation systems, such as advertising and personalized recommendations [10]. This algorithm also achieved crucial results in the financial and medical fields, such as balancing the risks and benefits of sequential portfolio selection problems and ambulance redeployment issues [11, 12]. In 2020, Trovò et al. introduced the SW-TS algorithm for non-stationary MAB problems. It unifies two non-stationary forms, has theoretical regret bounds, and outperforms state-of-the-art algorithms. Their research has contributed to solving dynamic allocation in non-stationary environments and provided a good guide for this study [13].

3 Methodology

3.1 Overview

The multi-armed bandit problem and its algorithm are the core part of the reinforcement learning field. They are widely used in dynamic decision-making situations, such as advertising, recommendation systems, and clinical trials. The primary algorithm components are arms and rewards. The algorithm continually chooses an action a_t among a set of actions Λ(arm). The selection of action $\in\Lambda$ in a trial t brings out a specific reward $R_t(a_t)\in \mathbb{R}$, which can be summarized as a real number. The core feature of the algorithm is to balance exploration and exploitation to obtain the maximum cumulative reward. If the agent only explores, it selects actions randomly, disregarding prior knowledge. Conversely, if it only exploits, it focuses on immediate rewards—similar to greedy methods—and may miss long-term benefits. The algorithm has changed strategy, switching from maximizing reward to minimizing regret, which is presented as formula 1. In this formula, $r_t^*$is the most enormous mean reward among all arms, and r_t is the reward corresponding to a certain arm. Regret has also become a vital evaluation metric for multi-armed bandit algorithms. Generally speaking, the smaller the cumulative regret, the closer the algorithm is to the optimal strategy.

$$Maximize \sum_{t=1}^{T} r_t \equiv Minimize \sum_{t=1}^{T} (r_t^* - r_t) \tag{1}$$

The overall framework of the multi-armed bandit algorithm will be shown in Figure 1.

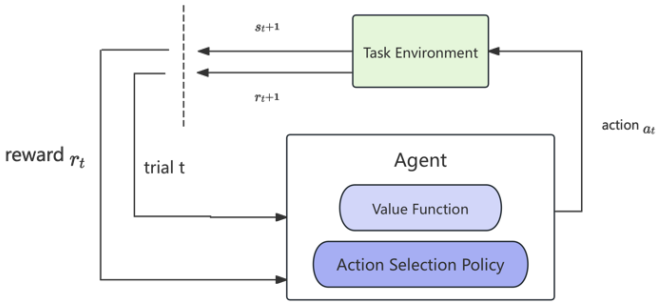

Figure 1 The framework of Multi-Armed Bandits (Photo credit: Original)

3.2 Data Processing

This study uses a public dataset from February 2024 released on the official website of Citi Bike in New York, USA, which contains more than 50,000 travel records and records 13 feature dimensions. This includes the user's membership information, user ID, vehicle type, the station name, ID, longitude, latitude of the starting and ending stations, travel time, and arrival time. The study read the data in Python from the pandas library. First, the study cleared all models except e-bikes. Then, the research calculated each trip record's travel time and distance from the starting station to the terminal station. In terms of riding distance, given the latitude φ_1, φ_2 and longitude λ_1, λ_2 of the starting station and the terminal station given in the data set, the study used the Haversine formula to calculate the distance on the Earth's surface between the two points. The Haversine formulas are as follows. Regarding riding

costs, the study set the income per kilometer and the cost loss per kilometer to 3 dollars and 0.5 dollars, respectively(the actual price will fluctuate within a specific range). Then, each starting station and its outliers in the data were cleaned up, such as the minimum travel time greater than 180 minutes and the minimum riding distance greater than 30 km.

$$a = sin^2\left(\frac{\Delta\varphi}{2}\right) + cos(\varphi_1) \cdot cos(\varphi_2) \cdot sin^2\left(\frac{\Delta\lambda}{2}\right) \tag{2}$$

$$d = 2R \cdot arcsin\left(2\sqrt{a}\right) \tag{3}$$

3.3 Game Setting

This experiment imports the filtered data into a new script, sets each starting station as an arm, and defines the expected reward as the average net profit corresponding to each starting station. If the game assumes that a certain starting station is selected m times, there will be the following formula：

$$E(r_t) = \frac{1}{m}\sum_{i=1}^{m} r_t \tag{4}$$

The game then iterates over the expected rewards for each arm and finds the optimal expected rewards r_t^*. Subsequently, the study set each arm as an index to facilitate subsequent experiments and plotted the expected reward of each starting station using the matplotlib library for observation in subsequent experiments, as shown in Figure 2.

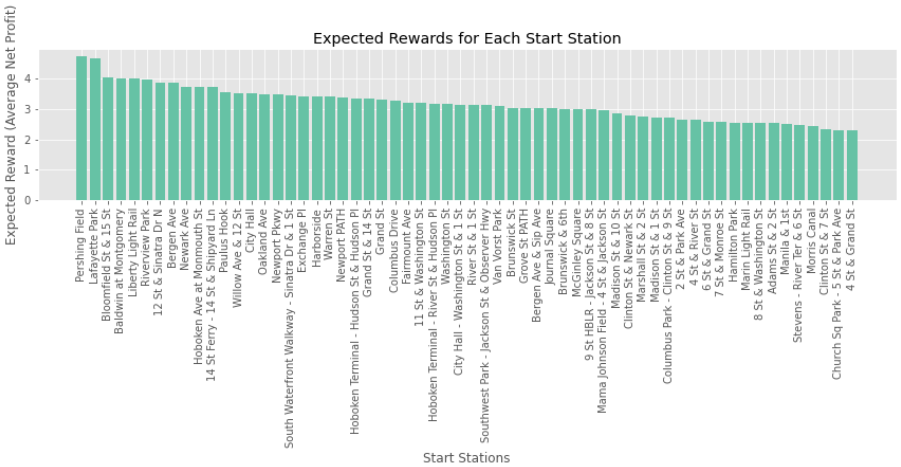

Figure 2 Expected Rewards for Each Start Station (Photo credit: Original)

3.4 Introduce The Baseline

3.4.1 UCB

The UCB algorithm is one of the most classic algorithms in the multi-armed bandit problem. It is dedicated to balancing the conflict between "exploration" and "exploitation." Its core idea is to select the arm with the highest upper confidence bound in each decision. This means

not only looking at the current average reward but also considering the uncertainty of knowledge about that arm. The core formula of the UCB algorithm is as follows:

$$UCB_i(t) = \bar{x}_i(t-1) + \frac{B}{2}\sqrt{\frac{4\,log\,n}{N_i(t-1)}} \tag{5}$$

$UCB_i(t)$represents the upper confidence bound of the i-th arm at the t-th decision point, $\bar{x}_i(t-1)$ is the average reward (sample mean) obtained by the i-th arm so far, t is the current total number of trials and $N_i(t-1)$ is the number of times the i-th arm is selected. $logn$ is the natural logarithm, ensuring that the confidence intervals scale with the number of trials. $\sqrt{\frac{4\log n}{N_i(t-1)}}$ is the confidence interval width, which is used to quantify the uncertainty of the reward estimate for that arm. When the average reward $\bar{x}_i(t-1)$ of an arm is high, the UCB algorithm will be more inclined to choose it (exploit). When $N_i(t-1)$ is small, the second term is large, indicating that the algorithm does not know enough about the arm and should explore more. The algorithm makes decisions by selecting argmaxUC$B_i(t)$ and choosing the arm with the highest upper confidence bound to maximize the expected return while considering the underexplored arms.

3.4.2 TS

The core of the Thompson sampling algorithm lies in probability matching. At each time step, the algorithm optimizes the strategy by selecting the action a with the highest probability. The optimal strategy is based on the currently known action power h_t, and its form is:

$$\begin{aligned}\pi(a|h_t) &= \mathrm{P}[Q(a) > Q(\acute{a}), \forall \acute{a} \neq a|h_t] \\ &= E[a = argmax_{a\in A}Q(a)]\end{aligned} \tag{6}$$

Modern Thompson sampling algorithms usually assume that the Q function follows a Bernoulli distribution Beta distribution, with the probability of success represented by n. Specifically, the value of the Beta distribution Beta(α,β) is in the interval [0,1], where the parameters α and β correspond to the counts of reward gains and failures, respectively. At each time point t, the agent must sample the expected reward $\tilde{Q}$(a) for each action a from the prior distribution Beta(α_i,β_i). Among all samples, the best action is selected as:

$$a_t^* = argmax_{a\in A}\tilde{Q}(a) \tag{7}$$

Once the valid reward is observed, the Beta distribution is updated accordingly through Bayesian inference. If the selected action is correct (i.e., $a_t^* = a_i$), then α will be updated; otherwise, β will be increased. This method performs well in various practical applications [9].

3.5 Sliding-Window Thompson Sampling

Regarding algorithmic principles, SW-TS is an innovative algorithm specifically for the non-stationary multi-armed bandit problem. It mainly deals with two types of non-stationarity: the first is Abruptly Changing, which means that the reward distribution remains unchanged in some rounds but may suddenly change in unknown rounds. The second is Smoothly

Changing, meaning the reward distribution evolves smoothly according to unknown dynamics. The SW-TS algorithm still uses the Bayesian method, and its basic process is based on the Thompson algorithm, so this part of the architecture is not described here. A key feature of the SW-TS algorithm is introducing the concept of a sliding window mechanism; that is, the algorithm uses a sliding window of a fixed length τ. The purpose of this is only to consider the reward information of the most recent τ rounds and to reduce the bias of historical data on decision-making by "forgetting" past information; that is, historical information only uses information within the sliding window. In this way, it effectively optimizes the algorithm in non-stationary situations. When faced with some stable changes or sudden changes, the information obtained by the algorithm is more reliable and timely, and has strong practical significance. The core formula is as follows：

$$\pi_{i,t} = Beta\left(S_{i,t,\tau} + 1, T_{i,t,\tau} - S_{i,t,\tau} + 1\right) \tag{8}$$

where $S_{i,t,\tau}$ refers to the number of successes of arm i in the last τ rounds, and $T_{i,t,\tau}$ refers to the total number of times arm i has been pulled in the previous τ rounds. Generally speaking, the theoretical regret bounds in abrupt change environment and a smooth change environment are $O(N^{\frac{1+\alpha}{2}})$ and $O(N^{\beta})$ respectively. α is used to measure the number of mutations, and β is used to measure the proximity of rewards.

3.6 Evaluation Metrics

3.6.1 Cumulative Reward

Cumulative reward refers to the total reward obtained by the algorithm during the entire decision-making process in the multi-armed bandit problem, and its formula is as follows：

$$R(T) = \sum_{t=1}^{T} r_t \tag{9}$$

R(T) is the cumulative reward after T rounds, and r_t is the immediate reward obtained by the arm selected in round t.

3.6.2 Cumulative Regret

Cumulative regret measures the expected reward lost by the algorithm choosing a non-optimal arm, and its formula is as follows:

$$Regret(T) = n * r_t^* - \sum_{t=1}^{T} E(r_t) \tag{10}$$

r_t^* is the expected reward of the best arm, and r_t is the expected reward of the arm actually selected by the algorithm in round t.

3.6.3 Click-Through Rate(CTR)

In the multi-armed bandit recommendation system, the click-through rate refers to the probability that the recommended item is clicked.

$$CTR = \frac{n(t)}{n} \tag{11}$$

n(t) refers to the number of clicks, and n refers to the number of impressions.

3.6.4 Comparison with Actual Reward and Expected Reward

This evaluation index theoretically reflects the algorithm's learning efficiency, quantifies its uncertainty, and evaluates the accuracy of probability estimation. In terms of practical value, it can diagnose the algorithm's robustness in different scenarios and optimize the exploration-exploitation balance strategy.

4 Experiments and Results

4.1 Experimental Settings

The study used the UCB, TS, and SW-TS algorithms to conduct experiments according to the above evaluation metrics to measure the effect of the algorithm on the dataset and provide the optimal solution for the dynamic allocation of e-bikes. Before the experiment, the study used filtered data to calculate the reward range for the B value of the UCB formula. Since this dataset is still static data, this study added non-stationary conditions to the SW-TS experiment. In view of some uncontrollable factors such as station maintenance and the particularity of station location (around schools, around governments), some stations may not be used every day, so this study introduced abrupt changes. The following are some disruptions for the dynamic environment of the SW-TS experiment, as shown in Table 1.

Table 1 Setting for the dynamic environment

Time	Event_type	intensity
5000	reward_drop	0.5
10000	reward_spike	1.5
15000	high_variance	2.0

At the same time, the seasonal factor is introduced, and the formula is as follows：

$$seasonal\ factor = sin\left(\frac{timestamp}{5000}\right) * 0.2 + 1 \tag{12}$$

Therefore, the reward for non-stationary event processing is shown in Figure 3.

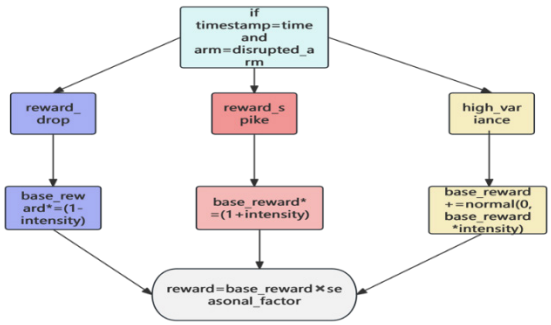

Figure 3 The reward for non-stationary environments (Photo credit: Original)

4.2 Results

4.2.1 Results of Cumulative Regret

In this experiment, the total number of rounds for UCB, TS, and SW-TS was set to 30,000, and the experiment was repeated ten times. Error bars are used to indicate the range of one standard deviation above and below the mean. The sliding window size of the SW-TS algorithm is set to 0.005 of the total number of experiment rounds. The results are shown in Figures 4, 5, and 6.

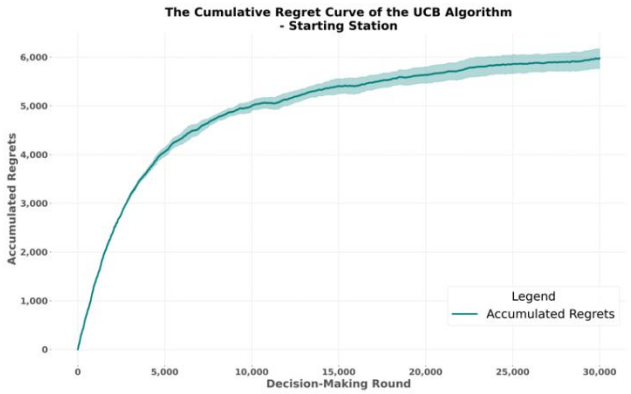

Figure 4 The cumulative regret of the UCB (Photo credit: Original)

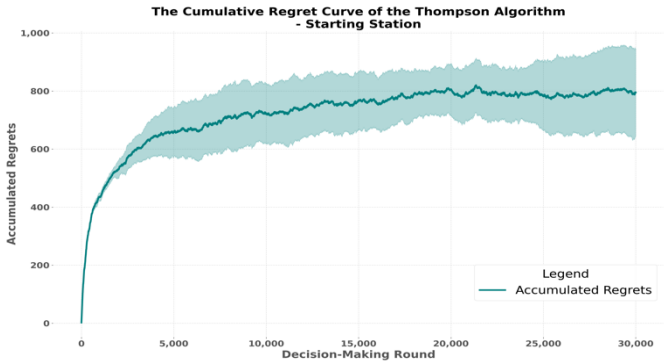

Figure 5 The cumulative regret of the TS (Photo credit: Original)

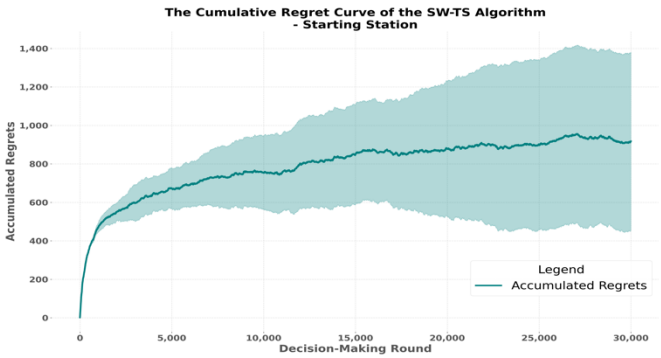

Figure 6 The cumulative regret of the SW-TS (Photo credit: Original)

Through the above image analysis, the study found that the regret of TS and SW-TS is much smaller than that of UCB, which proves that on this data set, the first two algorithms perform better in terms of regret and converge to the optimal arm faster.

4.2.2 Results of the Click Rate of the Optimal and Suboptimal Arms

In this experiment, each algorithm continued the total number of discussions and repetitions of the previous experiment. After experimental verification, the optimal arm corresponding to this dataset is Pershing Field, with an expected reward of 4.73 dollars; the second-best arm is Lafayette Park, with an expected reward of 4.65 dollars. The experiment obtained the corresponding click rates of the optimal arm and the suboptimal arm, as shown in Table 2.

Table 2 Comparison of click rates of the optimal and suboptimal arms of three algorithms

Click rate	UCB	TS	SW-TS
Pershing Field	66.2%	86.7%	93.0%
Lafayette Park	20.4%	11.8%	5.0%

The experimental results show that the click rates of UCB, TS, and SW-TS for the rightmost arm increase in turn, and the click rate of SW-TS for the optimal arm is as high as 93%. This indicates that the algorithm has a better application effect on this dataset, proving that it is better than the current baseline model in terms of decision quality, statistical learning ability, and information utilization.

4.2.3 Results of Comparison with Actual Reward and Expected Reward

This experiment quantifies the comparison between the algorithm's expected reward and actual reward into two values: MAE and Mean Squared Error (MSE). The specific formula is as follows:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{13}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{14}$$

This experiment took the MAE and MSE comparison of the actual and expected rewards of the top ten arms. The results of the experiment are shown in Table 3.

Table 3 Comparison of MAE and MSE

	UCB	TS	SW-TS
MAE	0.0865	0.2585	0.2188
MSE	0.0184	0.1611	0.1126

Unlike the previous two experiments, the UCB algorithm outperforms the other two algorithms in both MAE and MSE, which reflects the prediction accuracy. However, given that the MAE and MSE values of TS and SW-TS are also within a reasonable range, it can be concluded that these algorithms have a good effect on predicting the actual reward and expected reward for this dataset.

5 Conclusion

Through the above research, the researchers found that TS still performs well in cumulative regret, and UCB performs well in MAE and other indicators, proving that the multi-armed bandit algorithm still performs well in dealing with e-bike allocation optimization problems. In addition, SW-TS performs very well in coping with non-stationary environments, and its performance in regret, optimal arm click-through rate, MAE, and other metrics is mostly better than the existing baseline model, which plays a vital role in solving dynamic distribution optimization problems. Research shows that Citi Bike can put more e-bikes in stations such as Pershing Field, Lafayette Park, 12th ST & Sinatra Dr N, Bloomfield St & 15 St, and Van Vorst Park while promoting the increase in yield and perfecting the current research on dynamic allocation of e-bikes. This research has a specific indicative role in cities around the world that need to allocate station-based e-bikes reasonably, and it also proves that the multi-armed bandit algorithm still has strong strength in the field of resource push. Admittedly, the research still lacks some details, such as not considering special influencing factors, such as weather and holidays, as contextual features, and not using multiple months of data for experiments. Future experiments will improve these shortcomings, promote the solution to be more reasonable, and contribute to solving the dynamic allocation problem of station-based e-bikes.

References

1. Bourne, J.E., Sauchelli, S., Perry, R., Page, A., Leary, S., England, C., and Cooper, A.R.: 'Health benefits of electrically-assisted cycling: a systematic review', Int. J. Behav. Nutr. Phys. Act., 2018, 15, pp. 1-15.
2. Tarroja, B., Forrest, K., Yamada, K., Saha, R., and Hyland, M.: 'Estimating the Electricity System Benefits of Scaling up E-Bike Usage in California', J. Clean. Prod., 2025, (144840).
3. Lou, L., Li, L., Yang, S.B., and Koh, J.: 'Promoting user participation of shared mobility in the sharing economy: Evidence from Chinese bike sharing services', Sustainability, 2021, 13, (3), pp. 1533.
4. Fu, C., Zhu, N., Pinedo, M., and Ma, S.: 'Station-based, free-float, or hybrid: An operating mode analysis of a bike-sharing system', Transp. Res. Part B Methodol., 2025, 191, pp. 103105.

5. Luo, H., Kou, Z., Zhao, F., and Cai, H.: 'Comparative life cycle assessment of station-based and dock-less bike sharing systems', Resour. Conserv. Recycl., 2019, 146, pp. 180-189.
6. Hua, M., Yu, X., Chen, X., Chen, J., and Cheng, L.: 'Can bike sharing achieve self-balancing distribution? Evidence from dockless and station-based cases', Travel Behav. Soc., 2025, 38, pp. 100879.
7. Angelelli, E., Chiari, M., Mor, A., and Speranza, M.G.: 'A simulation framework for a station-based bike-sharing system', Comput. Ind. Eng., 2022, 171, pp. 108489.
8. Jara-Díaz, S., Latournerie, A., Tirachini, A., and Quitral, F.: 'Optimal pricing and design of station-based bike-sharing systems: A microeconomic model', Econ. Transp., 2022, 31, pp. 100273.
9. Silva, N., Werneck, H., Silva, T., Pereira, A.C., and Rocha, L.: 'Multi-armed bandits in recommendation systems: A survey of the state-of-the-art and future directions', Expert Syst. Appl., 2022, 197, pp. 116669.
10. Srikanth, A., Gowthaam, G., Gayathri, M., Goutham, D.T., Lakshana, C.R., and Lavaniya, H.: 'Dynamic personalized ads recommendation system using contextual bandits'. Proc. Int. Conf. Intelligent Systems for Communication, IoT and Security (ICISCoIS), Feb. 2023, pp. 339-344.
11. Huo, X., and Fu, F.: 'Risk-aware multi-armed bandit problem with application to portfolio selection', R. Soc. Open Sci., 2017, 4, (11), pp. 171377.
12. Şahin, Ü., Yücesoy, V., Koç, A., and TEKin, C.: 'Risk-averse ambulance redeployment via multi-armed bandits'. Proc. 26th Signal Processing and Communications Applications Conf. (SIU), May 2018, pp. 1-4.
13. Trovo, F., Paladino, S., Restelli, M., and Gatti, N.: 'Sliding-window thompson sampling for non-stationary settings', J. Artif. Intell. Res., 2020, 68, pp. 311-364