Abstract. Bike-sharing system has become an indispensable element of sustainable urban transportation, effectively resolving the “last mile” transportation challenge for city dwellers. A major daily operational task in these systems is planning a fleet to rebalance the bikes over time, ensuring the optimal availability of bikes and docks to users. Recycling is also a daily work with the increase in the number of broken bikes. However, rebalancing or recycling operation is always regarded as an independent task. They are separately studied in existing papers. Thus, this paper develops an operational strategy for recycling broken bikes during the rebalancing process, and studies the combination of the station inventory and vehicle routing problems. First, an inventory routing model is constructed with the aim of minimizing the total costs including procurement, expected user loss, inventory and transportation costs. Then, a two-stage iterative algorithm is developed with both exact and heuristic algorithms. We use real-world data from Capital Bikeshare to test our proposed model and approach, which shows the two-stage iterative algorithm is efficient and outperforms existing solutions in reducing total costs. Finally, the sensitivity analysis is performed on key parameters such as the vehicle’s capacity, unit penalty costs for customer dissatisfaction events, unit inventory holding costs and the observation period of rebalancing. It shows that enterprises can reduce the total cost by altering vehicle’s capacity, reducing the unit inventory holding costs or changing the observation period of rebalancing.

Mathematics Subject Classification. 35J20, 35J25, 35J60.

Received December 7, 2023. Accepted May 10, 2024.

1. Introduction

Micro-mobility options rise from urbanization, technological advancements, and the need for last-mile connectivity [1]. Bike-sharing is a crucial component of micro-mobility initiatives. The rapid expansion of bike-sharing services not only offers convenience to urban residents but also assists cities in alleviating traffic congestion while promoting a transition towards low-carbon modes of daily travel [2, 3]. As a new type of green public trans-
portation, the bike-sharing system provides services for public areas such as bus stations, subway entrances and commercial areas [4,5]. According to the annual report from The Meddin Bike-sharing World Map\(^1\), there were 8,967,122 shared bikes available across 1914 bike-sharing systems worldwide in August, 2022, covering 1,590 cities in 92 countries across all continents.

A series of interesting optimization problems arise within bike-sharing systems and other micro-mobility systems, such as rider behavior [6], station placement [7], and the rebalancing problem [1,8]. Notably, the bike-sharing rebalancing problem (BRP) has garnered significant attention in academic circles, aiming to address the spatial and temporal disparities in demand distribution through the allocation and relocation of bikes [9]. BRP can be divided into two types: static BRP and dynamic BRP [10]. In the static scenario, rebalancing operations occur in batches at scheduled times, such as during lunchtime or nighttime each day. In the dynamic case, adjustments are made flexibly at any time to meet fluctuating demand. Most of the research works, including this paper, focus on the static BRP.

To solve the BRP, inventory levels of stations and vehicle routes has been extensively studied separately. Wang et al. [11], Maggioni et al. [10], He et al. [12] and Harikrishnakumar et al. [13] determined inventory levels by forecasting future customer demand in stations. Besides, Schuijbroek et al. [14], Haider et al. [15], Vishkai et al. [16], and Swaszek et al. [17] treated the bike-sharing system as a queuing network for the study on inventory levels proved by Gast et al. [18]. We also employ queuing theory to ascertain inventory levels, but with different objectives. Schuijbroek et al. [14] and Haider et al. [15] all set a certain level of service to seek upper and lower bounds on inventory levels. Moreover, Datner et al. [19] set the inventory levels by minimizing the travel time users spent in the bike-sharing system. Swaszek et al. [17] minimized the expected costs over an interval of time from the users’ perspective to obtain inventory levels. This paper is a natural extension of this setting in that we minimize total costs considering expected accumulated user loss costs, procurement costs, and inventory costs. Furthermore, Vishkai et al. [16] considered interactions between stations and solving inventory levels by minimizing the ratio of dissatisfied customers and computed the solution. Possani and Castillo [20] also considers data from a public bike-sharing system, and solves the problem with interaction between stations. This article does not consider interaction between stations and regard each station as an independent queuing system. Moreover, both we and Guo et al. [21] take customer satisfaction into account.

Due to the allocation of inventory at the station, dispatching vehicles to relocate bikes is necessary. Many scholars, as well as this paper, conduct research on route planning for transport vehicles. Ren et al. [22] considered transportation costs including vehicle inventory costs and extended from the single-vehicle traveler problems to the multiple-vehicle route planning. The operator in this paper is supposed to have a fleet of transport vehicles. Du et al. [23], Zheng et al. [24] and Akova et al. [25] studied routes on heterogeneous trucks of different capacities. But a homogeneous fleet of transport vehicles is given priority in this paper. The NP-hard complexity of the BRP has prompted the development of various heuristics for tackling them [26]. Several different kinds of local search algorithms on vehicle routing planning mentioned by Brinkmann et al. [27]. Shi et al. [2] developed an improved particle swarm optimization algorithm. Although particle swarm optimization algorithm is simple and low in complexity, its diversity is lost, and it is likely to fall into local solutions. Ren et al. [28] designed an improved general variable neighborhood search algorithm to find better solutions across multiple neighborhoods, but it blindly explores each operator’s neighborhood structure. Pan et al. [29] compared capacity range length heuristic with tabu search algorithm, which had a strong local development ability with weak global development ability. However, the adaptive large neighborhood search (ALNS) algorithm used in this paper can fill these gaps of the aforementioned algorithms. ALNS algorithm can automatically select good operators to destroy and repair the solution, so that it is more likely to obtain a better solution.

More importantly, only a few studies on BRP address both inventory and routing aspects. Caggiani et al. [30] and Swaszek and Cassandras [17] discussed them separately, which is not conducive to reducing costs in the supply chain. One of the highlights of the article is the research conducted on the inventory routing problem in the bike-sharing system. Schuijbroek et al. [14] proposed a heuristic algorithm based on clustering-first

\(^1\) https://bikesharingworldmap.com/
and route-second by combining inventory and route. We utilize a two-stage iterative algorithm for solving the inventory routing problem, like Paeizi et al. [31]. Haider et al. [15] modified the price vector to fuse inventory and route planning problems. Our model takes transportation costs as a variable to connect inventory and routing problems.

In addition to studying BRPs, we also simultaneously considered recycling broken bikes. Because when the broken bikes are scattered, it becomes highly inefficient to arrange separate vehicles for recycling. As a result, it makes sense to consider recycling broken bikes at the same time in the rebalancing process. Recycling broken bikes is still vital to ensure the smooth operation of the bike-sharing systems. With an average economic life of three years, shared bikes are exposed to various weather conditions and undergo considerable workload, leading to their possible breakdown [32, 33]. Recycling broken bikes is beneficial to form a closed-loop management of the whole life cycle of shared bikes in a green and environmentally friendly state and improve the utilization rate of social resources [34]. The recycling of shared bikes has also gradually been paid attention to by the academic community. Most literature on recycling shared bikes examines the routes for recycling from the perspective of actual operation, such as Chang et al. [32], Chen et al. [35], Wang and Szeto [36] and Lu et al. [37]. Du et al. [23] and Zhang et al. [38, 39] also studied the routes for recycling and rebalancing. Our research problem is similar to that of Alvarez-Valdes et al. [8]. We both consider recycling broken shared bikes in the rebalancing process from inventory and routing two aspects. However, Alvarez-Valdes et al. [8] set inventory levels by proposing an approximate method to predict the unsatisfactory demand of the station indirectly and this method could not guarantee the accuracy of the results. We use the finite birth and death process to simulate the demand fluctuates of each station. The unsatisfactory demand is predicted directly by the finite birth and death process and the results are more accurate.

Building on the preceding analyses, this paper starts by examining each dissatisfaction event from customers at every station and studies the inventory routing problem for the BRP with recycling broken bikes. This results in the construction of a mixed integer non-linear programming model that incorporates the expected accumulated user loss cost. Considering that the problem addressed in this paper is complex, and solving static problems effectively is the only way to further solve more complex dynamic rebalancing problems. Hence, this paper will address the static BRP. We propose a two-stage iterative algorithm to solve the inventory routing problem in this paper. The first stage focuses on the inventory rebalance subproblem, but vehicle capacity constraints are taken into account during this stage. This stage also takes a station-dependent inventory policy based on demand rates and station capacity and determines the quantity of bikes that need to be handled at stations. The second stage concentrates on route planning, aiming to optimize routes with minimal transportation costs. After linearizing the model using dummy variables, we design an ALNS algorithm with Metropolis acceptance criterion. Furthermore, in the first stage, we obtain routing data by assigning an access cost to stations, which is then updated in each iteration based on the results of the second stage. Finally, the algorithm’s efficiency is evaluated through a set of instances. The main contribution to the literature is threefold.

(1) Few studies have considered the combination of the BRP and recycling broken bikes. It is more in line with the current development of the urban shared bike industry [35]. This paper introduces a novel variation of the BRP that fills this gap and has theoretical significance.

(2) This study aims to minimize total costs by optimizing inventory levels at the station and routes of a fleet simultaneously. We take total costs into account comprehensively, including the expected accumulated user loss, procurement, bike inventory, and transportation costs. This study also enriches the research on the inventory routing problem in bike-sharing systems and provides references for other micro-mobility systems.

(3) A two-stage algorithm consisting of an exact algorithm and a heuristic algorithm is developed in this paper. In routing planning subproblem, the ALNS algorithm with tailored destroy and repair operators is designed to obtain the final solution efficiently. Real data from Capital Bikeshare is used to test and verify the feasibility of the two-stage algorithm of this paper. The computation results show that the integrated inventory

http://www.capitalbikeshare.com/
routing decisions we proposed outperforms both independent recycling and decentralized decisions regarding inventory and routing in terms of total costs.

The remainder of the paper is organized as follows. Section 2 describes the proposed problem in detail, defines the sets, parameters, and variables, and then constructs the model. Section 3 elaborates the processes and steps of the two-stage iterative algorithm. In Section 4, we conduct numerical experiments to demonstrate the properties of the proposed model and validate the efficiency of the proposed algorithm, and sensitivity analysis is also conducted. Finally, we conclude this study and point future research prospects in Section 5.

2. INVENTORY ROUTING MODEL

We consider a bike-sharing system with $n$ stations in set $B = \{1, 2, 3, \ldots, n\}$, a distribution center, and transport vehicles in set $K = \{1, 2, 3, \ldots, F\}$. Each station has a set number of parking spaces and shared bikes, as well as a limited capacity. Customers arrive at each station randomly to rent or return a bike. But once they are unable to rent or return a bike at the location, their experience with biking will be negatively impacted. These are called opportunity loss. We assign a penalty for each customer dissatisfaction event, respectively, when a rent or return demand cannot be satisfied. And the total of penalties over period $[0, t]$ is expected accumulated user loss costs, similar to Delivand et al. [40]. The allocation of bikes in the bike-sharing system is constantly shifting due to diversified riding demand and customers’ random behavior. Broken bikes are unavoidable in the system due to the influence of multiple factors. In addition, we suppose the rebalance operation takes place at noon or night. Because this paper studies static shared bike rebalancing problems, and the number of shared bikes at noon or night changes small which can be ignored. At lunchtime or at nighttime every day, each station is routinely evaluated for the number of broken bikes and available bikes. The results of the monitoring are used as the foundation for rebalancing and recycling. The vehicle routing problem in our paper can be viewed as a pickup and delivery vehicle routing problem, since transport vehicles may involve picking up or recycling bikes at each station or delivering bikes.

The distribution center can only send out transport vehicles each with a capacity of $Q$ bikes each time to serve as a general warehousing and recycling facility for shared bikes. The distribution center is both the starting and ending depots of all transport vehicles. Transport vehicles recycle broken bikes from each station and move surplus usable bikes from certain stations to other stations that need them. The primary expense associated with the process of rebalancing and recycling is the transportation cost. Transportation costs consist of setup costs and fuel costs. The setup cost covers the cost of dispatching a vehicle, including driver salaries and depreciation. Moreover, the distribution of shared bikes and parking space at each station have a direct impact on the probability of empty or filled stations. Thus, the new distribution of shared bikes after rebalancing and recycling will affect the expected user loss costs by impacting subsequent user satisfaction. Additionally, each shared bike parked in the station or distribution center requires an equal share of space, management personnel, and other costs, which are categorized in this paper as unit inventory cost.

Our goal is to minimize the total costs, consisting of procurement, expected user loss, inventory and transportation costs. The optimal inventory level, optimal number of transport vehicles used and the optimal route are solved under the presumption that the inventory capacity of each station and the capacity constraints of each transport vehicle are satisfied. The optimal inventory level is the inventory level that should be held at each station.

The model parameters, decision variables, and sets are shown in Tables 1–3.

Objective function

$$\min c_1 + c_2 Q + \sum_{i \in B} \tau_i (I^i_0, t) + hQ + \sum_{k \in K} \sum_{j \in B} f_k x^k_{0j} + \sum_{k \in K} \sum_{i \in V_0, j \in V_{n+1}, i \neq j} (E_k + e_k w_i^k) d_{ij} x^k_{ij}. \quad (1)$$

Subject to

$$Q = \sum_{i \in B} I^i_0 \quad (2)$$
TABLE 1. Model sets.

<table>
<thead>
<tr>
<th>Sets</th>
<th>Set meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>$dc_0$</td>
<td>Start point</td>
</tr>
<tr>
<td>$dc_{n+1}$</td>
<td>End point</td>
</tr>
<tr>
<td>$B$</td>
<td>Set of stations, $B = {1, 2, 3\ldots, n}$</td>
</tr>
<tr>
<td>$V_0$</td>
<td>$V_0 = dc_0 \cup B$</td>
</tr>
<tr>
<td>$V_{n+1}$</td>
<td>$V_{n+1} = dc_{n+1} \cup B$</td>
</tr>
<tr>
<td>$K$</td>
<td>Set of transport vehicles, $K = {1, 2, 3\ldots, F}$</td>
</tr>
</tbody>
</table>

TABLE 2. Model parameters.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Parameter meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>$c_1$</td>
<td>Setup cost of procurement</td>
</tr>
<tr>
<td>$c_2$</td>
<td>Unit price of a shared bike</td>
</tr>
<tr>
<td>$h$</td>
<td>Unit inventory cost of a bike in distribution center</td>
</tr>
<tr>
<td>$f_k$</td>
<td>Unit setup cost of a transport vehicle $k, k \in K$</td>
</tr>
<tr>
<td>$E_k$</td>
<td>Fuel cost generated by vehicle $k$ without load per unit distance, $k \in K$</td>
</tr>
<tr>
<td>$e_k$</td>
<td>Additional fuel cost generated by vehicle $k$ for one unit of cargo per unit distance, $k \in K$</td>
</tr>
<tr>
<td>$d_{ij}$</td>
<td>Distance of arc$(i, j)$, $i \in V_0, j \in V_{n+1}, i \neq j$</td>
</tr>
<tr>
<td>$r_i$</td>
<td>Number of broken shared bikes at station $i, i \in B$</td>
</tr>
<tr>
<td>$c_3$</td>
<td>Unit penalty occurred when a customer arrives at the station and cannot rent a shared bike</td>
</tr>
<tr>
<td>$c_4$</td>
<td>Unit penalty occurred when a customer arrives at the station and cannot find a vacant space to return the shared bike</td>
</tr>
<tr>
<td>$u_i$</td>
<td>Rental demand rate of station $i, i \in B$</td>
</tr>
<tr>
<td>$\lambda_i$</td>
<td>Return demand rate of station $i, i \in B$</td>
</tr>
<tr>
<td>$C_i$</td>
<td>Inventory capacity of station $i, i \in B$</td>
</tr>
<tr>
<td>$F$</td>
<td>Total number of transport vehicles</td>
</tr>
<tr>
<td>$S^i_0$</td>
<td>Current inventory level at station $i, i \in B$</td>
</tr>
<tr>
<td>$W_k$</td>
<td>Maximum load capacity of the vehicle $k, k \in K$</td>
</tr>
</tbody>
</table>

\[
\tau_i (I^i_0, t) = \int_0^t p_{i,0}(I^i_0, t)u_i c_3 + p_{1,c_i}(I^i_0, t)\lambda_i c_4, \quad \forall i \in B
\]  

\[
0 \leq I^i_0 \leq C_i, \quad \forall i \in B
\]  

\[
\sum_{k \in K} \sum_{j \in V_{n+1}} x^k_{ij} = 1, \quad \forall i \in B
\]  

\[
\sum_{k \in K} \sum_{i \in V_0} x^k_{ij} - \sum_{k \in K} \sum_{i \in V_{n+1}} x^k_{ji} = 0, \quad \forall j \in B
\]  

\[
\sum_{k \in K} \sum_{j \in B} x^k_{0j} \leq F
\]  

\[
g_i = I^i_0 - S^i_0, \quad \forall i \in B
\]  

\[
w^k_i = w^k_{1i} + w^k_{2i}, \quad \forall i \in V_0, k \in K
\]
The objective function (1) minimizes total costs by rebalancing the number of shared bikes at each station, recycling broken shared bikes, and planning vehicle routing. The total costs consist of four main components, namely procurement, expected accumulated user loss, inventory and transportation costs. The procurement cost includes the fixed procurement setup cost and the cost of shared bikes. Constraint (2) refers to the total number of shared bikes currently needed. Constraint (3) defines the expected accumulated user loss costs caused by incidents of customers dissatisfaction where the first term is related to the costs of customers unable to rent a shared bike and the second term calculates the costs incurred by customers who fail to find a vacant space to return a used shared bike. Constraint (4) ensures that the inventory level of each station is bounded by corresponding inventory capacity. Constraint (5) sets each station to be accessed only once. Constraint (6) is the flow balance constraint to ensure that vehicles do not linger at the station. Constraint (7) states that the number of vehicles used is no more than the total number of transport vehicles. Constraint (8) defines demand

<table>
<thead>
<tr>
<th>Decision variables</th>
<th>Decision variable meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>$I_0^i$</td>
<td>Optimal inventory level of station $i$ at the present, $i \in B$</td>
</tr>
<tr>
<td>$Q$</td>
<td>Total number of shared bikes required at present</td>
</tr>
<tr>
<td>$\tau_i (I_0^i, t)$</td>
<td>The expected accumulated user loss cost at station $i$ after $t$ time units given an initial inventory level $I_0^i$, $i \in B$</td>
</tr>
<tr>
<td>$p_{i,X} (I_0^i, t)$</td>
<td>Transient state probability when inventory of station $i$ is $X$ after $t$ time units given an initial inventory level $I_0^i$, $i \in B$</td>
</tr>
<tr>
<td>$x_{ij}^k$</td>
<td>Binary variable indicates whether vehicle $k$ passes through arc$(i, j)$, $i \in V_0$, $j \in V_{n+1}$, $i \neq j$, $k \in K$</td>
</tr>
<tr>
<td>$q_i^k$</td>
<td>Auxiliary variable for subtour elimination inequalities, $i \in V_0$, $k \in K$</td>
</tr>
<tr>
<td>$w_{ki}$</td>
<td>Available bike load on vehicle $k$ leaving from node $i$, $i \in V_0$, $k \in K$</td>
</tr>
<tr>
<td>$w_{2i}$</td>
<td>Broken bike load on vehicle $k$ leaving from node $i$, $i \in V_0$, $k \in K$</td>
</tr>
<tr>
<td>$g_i$</td>
<td>Demand of station $i$, $i \in B$</td>
</tr>
</tbody>
</table>

\[ w_{1i}^k = \sum_{j \in B} g_j x_{ij}^k, \quad \forall i \in \{d_0\}, k \in K \]  
\[ w_{2i}^k = 0, \quad \forall i \in \{d_0\}, k \in K \]  
\[ w_{1j}^k \geq w_{1i}^k - g_j x_{ij}^k + (x_{ij}^k - 1)M, \quad \forall i \in V_0, j \in B, k \in K \]  
\[ w_{1j}^k \leq w_{1i}^k - g_j x_{ij}^k - (x_{ij}^k - 1)M, \quad \forall i \in V_0, j \in B, k \in K \]  
\[ w_{2j}^k \geq w_{2i}^k + r_j x_{ij}^k + (x_{ij}^k - 1)M, \quad \forall j \in B, k \in K \]  
\[ w_{2j}^k \leq w_{2i}^k + r_j x_{ij}^k - (x_{ij}^k - 1)M, \quad \forall j \in B, k \in K \]  
\[ w_{2j}^k \geq w_{2i}^k + r_j x_{ij}^k + \left( \sum_{k \in K} x_{ij}^k - 1 \right)M, \quad \forall i \in V_0, j \in B \]  
\[ w_{2j}^k \leq w_{2i}^k + r_j x_{ij}^k - \left( \sum_{k \in K} x_{ij}^k - 1 \right)M, \quad \forall i \in V_0, j \in B \]  
\[ q_i^k - q_j^k + |V_0| x_{ij} \leq |V_0| - 1, \quad \forall i \in V_0, j \in V_{n+1}, k \in K \]  
\[ 0 \leq w_i^k \leq W_i, \quad \forall i \in V_0, k \in K \]  
\[ x_{ij} \in \{0, 1\}, \quad \forall i \in V_0, j \in V_{n+1}, k \in K. \]
of station as the difference between the optimal inventory level and the current inventory level. Constraints (9)–(17) represent the number of shared bikes loaded on the vehicle. Constraint (9) indicates the total number of shared bikes loaded on the vehicle is composed of the number of shared bikes available and the number of broken shared bikes recycled. Constraints (10) and (11) display successively the number of available shared bikes and the number of broken shared bikes before departure of the transport vehicle. Constraints (12) and (13) define the number of remaining shared bikes available when the transport vehicle leaves all nodes except the destination. Constraints (14)–(17) refer to the total number of broken shared bikes recycled on the transport vehicle, namely the number of broken shared bikes recycled after leaving the first station and the total number of broken shared bikes in the transport vehicle routing. Constraint (18) are subcycle elimination inequalities, ensuring that each route of the vehicles passes through the start point, where \( q_k^i \) is an auxiliary variable representing the order of node visits on the route. Constraint (19) ensures that the total number of shared bikes on vehicle \( k \) leaving each station must be less than the maximum load capacity of the vehicle. Constraint (20) sets \( x_{ij}^k \) to be a binary variable.

3. Design of two-stage iterative algorithm

The inventory routing problem is NP-hard, because it includes classic vehicle routing problem [26]. The complexity of this problem makes it unable to be solved by any exact algorithm [41]. Therefore, this paper adopts a two-stage algorithm similar to that proposed by Absi et al. [42] for inventory routing problem. We split the model to reduce the difficulty of constructing the model. Below is the precise procedure. Firstly, we decompose the major problem into two subproblems: inventory rebalancing subproblem and route planning subproblem. The subproblems are solved by traversal method and ALNS algorithm respectively. Then, the two subproblems are combined while the transportation costs are updated using the integrated algorithm. Finally, the feasible solution with a minimum of the total costs is selected as the final solution of the original problem.

3.1. Inventory rebalance subproblem

This section describes the inventory rebalancing stage model and the algorithm steps that go with it. We consider both inventory and routing, and only solve the inventory rebalancing subproblem temporarily. Therefore, the corresponding connection between the two subproblems must be established. Access costs FC are used as a bridge connecting two subproblems in this paper, due to the fact that it affects the inventory rebalancing subproblem in calculating the optimal current inventory level. For more details about the use of access costs (FC), see Section 3.3. In order to ensure the completion of inventory rebalancing and recycling broken shared bikes at each station, this subproblem still requires that station capacity limits and transport vehicle loading constraints be satisfied. A straightforward approximation method is used to estimate transportation costs in this phase: FC is counted when each pickup and delivery is planned for a combination of vehicles and stations. The initial FC has the following calculation rules. To rebalance inventory in the bike-sharing system and carry broken bikes to the distribution center, one transport vehicle is set up for each station. After all stations have been visited by transport vehicles, access costs are computed. The starting value of FC is the sum of these transportation costs.

The inventory rebalancing phase model seeks to reduce the costs of procurement, expected accumulated user loss, inventory, and transportation associated with inventory rebalancing. According to the formula of the expected accumulated loss cost, the transition probabilities of each station’s inventory being empty and full after \( t \) time units are required to be calculated. Each station \( i \in B \) in bike sharing system can be treated as an independent \( M/M/1/C_i \) queuing system with limited system capacity. So the queuing system is a finite birth and death process (Fig. 1) subject to return demand rate \( \lambda_i \) (i.e. birth rate) and rental demand rate \( u_i \) (i.e. death rate).

Since the exchange between the limit of the transition probabilities and the sum of the corresponding formula in the birth and death process always holds, we are able to use the Kolmogrov forward equations to obtain the
following conclusions.

\[
\begin{align*}
\dot{p}_{i,0}(I_0^i, t) &= -\lambda_i p_{i,0}(I_0^i, t) + u_i p_{i,1}(I_0^i, t), & X = 0 \\
\dot{p}_{i,m}(I_0^i, t) &= \lambda_i p_{i,m-1}(I_0^i, t) - (\lambda_i + u_i) p_{i,m}(I_0^i, t) + u_i p_{i,m+1}(I_0^i, t), & X = 1, 2, \ldots, C_i - 1 \\
\dot{p}_{i,C_i}(I_0^i, t) &= \lambda_i p_{i,C_i}(I_0^i, t) - u_i p_{i,1}(I_0^i, t), & X = C_i.
\end{align*}
\]

These equations form a system of first order ordinary differential equations. We take advantage of the fourth order Runge–Kutta method solving this system of differential equations to obtain the transition probabilities. The fourth-order Runge–Kutta algorithm estimates the value of the function at the next point in time more accurately by calculating the slope at four different points.

Model construction for this subproblem is as follows.

Objective function

\[
\min c_1 + c_2 Q + \sum_{i \in B} \tau_i(I_0^i, t) + hQ + FC.
\]

Subject to (2)–(5), (8)–(11), (19)–(20)

\[
FC = \sum_{k \in K} \sum_{j \in B} f_k x_{0j}^k + \sum_{k \in K} \sum_{j \in B} (E_k + e_kw_j^k) d_{0j} x_{0j}^k + \sum_{k \in K} \sum_{j \in B} (E_k + e_kw_j^k) d_{jn+1} x_{jn+1}^k
\]

\[
x_{0j}^k - x_{jn+1}^k = 1, & \forall j \in B, k \in K \\
w_j^k \geq (u_j^k + r_j - g_j)x_{0j}^k + (x_{0j}^k - 1)M, & \forall j \in B, k \in K.
\]

The traversal method which is an exact algorithm can be used to resolve the inventory rebalancing problem. And the target value for the subproblem model under each state is obtained by adjusting the initial inventory of each station and using the state transition probability. The ideal target value is in line with the optimal inventory level of each station at the moment. The quantity of bikes taken and delivered at each station is then determined according to the optimal inventory level and the distribution of broken bikes at each station. The pseudo-code for the traversal algorithm is shown in Algorithm 1.
operators. In this study, four different types of destroy operators and four different types of repair operators are used. The number of iterations is met. The neighborhood structure of ALNS algorithm consists of destroy operators and repair operators.

Algorithm 1. Pseudo-code of traversal algorithm.

1: **Input** inventory capacity, rental rate, return rate at each station
2: **for all** $i \in B$
3: **for all** $S_0 = 1 : C_i$
4: Select time range $t$, set the time step $h$, and take $p_i, S_0(S_0, 0) = 1$, $p_i, X(S_0, 0) = 0(X \neq S_0)$ as the initial state
5: Use R-K4 method to solve the system of differential equations mentioned earlier
6: $K_1 = \hat{p}(t(k - 1), p(k - 1))$
7: $K_2 = \hat{p}(t(k - 1) + \frac{h}{2}, p(k - 1) + \frac{h}{2}K_1)$
8: $K_3 = \hat{p}(t(k - 1) + \frac{h}{2}, p(k - 1) + \frac{h}{2}K_2)$
9: $K_4 = \hat{p}(t(k - 1) + h, p(k - 1) + hK_3)$
10: $p(k) = p(k - 1) + \frac{h}{2}(K_1 + 2K_2 + 2K_3 + K_4)$
11: Obtain $p_{i,0}(S_0, t)$ and $p_{i,c}(S_0, t)$, and thus calculate $\tau_i(S_0, t)$ by constraint (3)
12: $S_0^* \leftarrow \text{Objective function (24)}$
13: **Output** $I_0^* = S_0^*$
14: **end for**
15: **end for**

3.2. Route planning subproblem

The route planning stage model resolves the simultaneous pickup and delivery vehicle routing problem for the purpose of minimizing transportation costs.

Objective function

$$\min \sum_{k \in K} \sum_{j \in B} f_k x_{0j}^k + \sum_{k \in K} \sum_{i \in V_0, j \in V_{n+1}, i \neq j} \left(E_k + e_k w_i^k\right) d_{ij} x_{ij}^k,$$

Subject to (5)–(20).

However, the proposed model is a nonlinear programming, since the objective function (28) contains nonlinear terms $w_i^k x_{ij}^k$. We reformulate the proposed model and derive its equivalent mixed-integer linear programming model. By introducing dummy variables $z_{ij}^k = w_i^k x_{ij}^k$, the objective functions can be translated as follows.

$$\min \sum_{k \in K} \sum_{j \in B} f_k x_{0j}^k + \sum_{k \in K} \sum_{i \in V_0, j \in V_{n+1}, i \neq j} \left(E_k x_{ij}^k + e_k z_{ij}^k\right) d_{ij},$$

Regarding the use of dummy variable $z_{ij}^k$, the following additional constraints should be added into the set of original constraints:

$$z_{ij}^k \geq (x_{ij}^k - 1)M + w_i^k, \quad \forall i \in V_0, j \in V_{n+1}, k \in K \quad (30)$$
$$z_{ij}^k \geq 0, \quad \forall i \in V_0, j \in V_{n+1}, k \in K. \quad (31)$$

Formally, we derive the mixed-integer linear programming model which has been validated with Cplex solver as follows.

$$\min \sum_{k \in K} \sum_{j \in B} f_k x_{0j}^k + \sum_{k \in K} \sum_{i \in V_0, j \in V_{n+1}, i \neq j} \left(E_k x_{ij}^k + e_k z_{ij}^k\right) d_{ij}.$$  

Subject to (5)–(20), (30) and (31).

The route planning stage problem is a combinatorial optimization, which belongs to NP-hard. Exact algorithms are not suitable for solving large-scale examples of such problems. Consequently, we take the ALNS algorithm in this section. ALNS generates an initial solution to the problem and iterates until the desired number of iterations is met. The neighborhood structure of ALNS algorithm consists of destroy operators and repair operators. In this study, four different types of destroy operators and four different types of repair operators
are designed. ALNS algorithm selects the operator according to the historical performance and usage times of operators and generate the neighborhood structure by using the competition between the operators. The pseudo-code of the ALNS algorithm is shown in Algorithm 2.

Algorithm 2. Pseudo-code of ALNS algorithm.
1: **Input** demand, initial temperature $T$, temperature cooling rate $t$ and parameters
2: Obtain initial feasible solution $X_0$ generated by $K$-means clustering algorithm
3: The initial feasible solution is set as the current solution: $X = X_0$ and the current solution is set as the optimal solution: $X^* = X$
4: Initialize the weights of destroy operators and repair operators: $dw = (1, \ldots, 1), rw = (1, \ldots, 1)$
5: repeat
6: Initialize the scores of destroy operators and repair operators: $ds = (1, \ldots, 1), rs = (1, \ldots, 1)$
7: Based on weights $dw$, a destroy operator $d$ is randomly selected with probability $P_d$ from the destroy operator set $D$, and then $d(X)$ is acquire
8: Based on weights $rw$, a repair operator $r$ is randomly selected with probability $P_r$ from the repair operator set $R$, and then $d(X)$ is repaired to get a new solution $X_{\text{new}} = r(d(X))$
9: Formula (25) is employed to calculate the objective of the new solution: $f(X_{\text{new}})$
10: if $f(X_{\text{new}}) < f(X)$ then
11: $X = X_{\text{new}}$
12: end if
13: if $f(X_{\text{new}}) < f(X^*)$ then
14: $X^* = X_{\text{new}}$
15: end if
16: if $f(X_{\text{new}}) > f(X)$ then
17: Metropolis criterion is utilized to accept the new solution with a certain probability $\rho$
18: if $\rho > \text{random number of interval (0, 1)}$ then
19: $X = X_{\text{new}}$
20: else
21: $X = X$
22: end if
23: end if
24: Update $ds$ and $rs$, $dw$ and $rw$
25: $T \leftarrow T \times t$
26: until the number of iterations is met
27: **Output** $X^*$, $f(X^*)$

3.2.1. Initial solution

Due to the limitation of transport vehicle capacity, the initial solution must satisfy the demand of each station. The following are the exact steps of the initial solution produced by the $K$-means clustering algorithm.

1. According to the ratio of the overall demand for collection and recycling at each station to the vehicle load capacity, the initial clustering quantity $K$ is established.
2. $K$ stations are selected randomly from all stations as the initial centroid, and the clustering capacity is set as the transport vehicle capacity.
3. All of the stations are sorted in descending order based on demand for picking-up and recycling, and each station is assigned to the appropriate cluster in turn. The following is the station assignment procedure. Firstly, the distance between each cluster center and the station is calculated. Then, the shortest distance is assigned from the station to the cluster, and whether the cluster capacity satisfies the capacity requirement is determined. The cluster is assigned to the station if the capacity restriction is satisfied; if not, the cluster with the least optimal distance is assigned to the station. Checking whether the clustering capacity meets its capacity restriction is also necessary. The previous steps are repeated until all stations have been assigned.
(4) After all stations are assigned, the center of gravity of each cluster is calculated and the center coordinates of each cluster are updated.

(5) Whether the difference between the original and the new cluster center coordinates exceeds the threshold is examined. If so, step (2) is proceeded to choose randomly the clustering center coordinates once more.

(6) Whether each cluster has a feasible route is checked to meet the vehicle capacity constraint. If not, the quantity of cluster is \( K = K + 1 \), and all the steps above are repeated. Otherwise, the starting solution is generated and the clustering results are stored.

3.2.2. Destroy operators and repair operators

After the initial solution is generated, the destroy operator ruins the domain structure. Then the visited station set and the unvisited station set are built. Four different types of destroy operators are designed: deletion of the longest route, deletion of the shortest route, random removal of a station, and random removal of routes. The neighborhood structure is rebuilt by repair operators, leaving no unvisited stations in the set. In our paper, four repair operators are proposed: greedy insertion in view of the transportation cost, greedy insertion considering the total driving distance, greedy insertion in light of total cost with perturbation, greedy insertion in view of the number of shared bikes loaded on the transport vehicle.

3.2.3. Operator selection

The decision of algorithm is based on the idea that each participant’s selection probability in roulette is inversely correlated with their fitness value and that the higher their fitness, the higher their chance of winning. In the iterative process, the weight \( W_i \) of each operator is dynamically adjusted on the basis of the changes in the quality score \( S_i \) of the solution. The scoring criteria for new solutions generated after each iteration can be classified into four categories.

(1) The new iteration’s solution outperforms the global optimal solution, score \( s_1 \).

(2) The new iteration’s solution is better than the current solution but worse than the global optimal, score \( s_2 \).

(3) The new iteration’s solution is not a brand-new one, score \( s_3 \).

(4) Despite being inferior to the present solution, the new iteration’s solution meets the Metropolis acceptance criteria, score \( s_4 \).

Where, \( s_1 \geq s_2 \geq s_3 \geq s_4 \).

The Metropolis acceptance criterion accepts inferior solutions with some probability [43]. The initial temperature is set to be \( T_0 \), the cooling rate is set to be \( t \) (\( 0 < t < 1 \)), and the temperature update at the end of each iteration is set to be \( T = T \times t \) for Metropolis acceptance criteria. Taking the minimization of the objective function as an example, the current solution is accepted with probability \( \rho = e^{-|C_c - C_0|/T} \) if the target value \( C_c \) corresponding to it differs from the target value \( C_0 \) relating to the first solution.

The formula for updating the operator weight is \( W_i = \Upsilon W_i + (1 - \Upsilon)S_i, \Upsilon \in (0,1), i = 1, 2, \ldots, |\Omega| \). \( \Upsilon \) is the reaction coefficient to determine the response degree of the algorithm to the operator performance variation. The initial score of each operator is set to 0, and the probability of operator \( i \) being selected is:

\[
P_i = \frac{w_i}{\sum_{k \in \Omega} w_k}, i = 1, 2, \ldots, |\Omega|.
\]

We use the coordinates of 5~30 stations, the transfer demand of each station as the integer generated randomly \([-1, 3]\), and the number of damaged shared bicycles at each station as the integer generated randomly \([0, 2]\) as the calculation examples. MATLAB R2018b was used to verify the ALNS algorithm. Meanwhile, the calculation examples are applied to CPLEX 12.10.0, and the results are compared with those of the algorithm to verify the validity of the ALNS algorithm (we set the longest running time of CPLEX as 3600 s). Table 4 shows the running results of each example in CPLEX and ALNS.
Table 4. The running results of each example in CPLEX and ALNS.

<table>
<thead>
<tr>
<th>No. of stations</th>
<th>CPLEX</th>
<th>ALNS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Transportation costs</td>
<td>Time</td>
</tr>
<tr>
<td>5</td>
<td>285.439</td>
<td>2.86 s</td>
</tr>
<tr>
<td>6</td>
<td>306.948</td>
<td>3.39 s</td>
</tr>
<tr>
<td>7</td>
<td>317.468</td>
<td>3.52 s</td>
</tr>
<tr>
<td>8</td>
<td>417.281</td>
<td>4.02 s</td>
</tr>
<tr>
<td>9</td>
<td>440.774</td>
<td>4.41 s</td>
</tr>
<tr>
<td>10</td>
<td>450.335</td>
<td>5.73 s</td>
</tr>
<tr>
<td>11</td>
<td>571.603</td>
<td>3600 s</td>
</tr>
<tr>
<td>12</td>
<td>704.906</td>
<td>3600 s</td>
</tr>
<tr>
<td>13</td>
<td>711.578</td>
<td>3600 s</td>
</tr>
<tr>
<td>14</td>
<td>741.558</td>
<td>3600 s</td>
</tr>
<tr>
<td>15</td>
<td>795.180</td>
<td>3600 s</td>
</tr>
<tr>
<td>16</td>
<td>866.379</td>
<td>3600 s</td>
</tr>
<tr>
<td>17</td>
<td>804.516</td>
<td>3600 s</td>
</tr>
<tr>
<td>18</td>
<td>810.016</td>
<td>3600 s</td>
</tr>
<tr>
<td>19</td>
<td>903.862</td>
<td>3600 s</td>
</tr>
</tbody>
</table>

3.3. Updating transportation costs

In the inventory rebalancing subproblem, FC is set by the approximation approach at the beginning as the bridge linking the two subproblems in this paper. The new transportation costs are obtained after the route planning subproblem solved. Therefore, FC standing for transportation costs needs to be updated. However, the two subproblems merged once may not get the optimal solution, and hence we design an integrated algorithm to iterate the two subproblems, searching for the optimal solution faster. The pseudo-code of the integrated algorithm is shown in Algorithm 3.


1: **Input** $X^*$, $X_{FC}$
2: Initial FC with the transportation costs of the initial optimal route solution is updated $X^*$
3: **repeat**
4: In the inventory rebalancing subproblem, the expected accumulated user loss cost of each station is sorted in descending order, and the station with the largest expected accumulated user loss cost is selected and then the inventory is adjusted accordingly
5: Calculate the expected accumulated loss user costs of each inventory of the station selected when it changes within the inventory constraint range of the station and sort the costs in descending order
6: Find out the position of the expected user accumulated loss function homologous inventory of the selected station, choose the homologous inventory below it next to the value of the expected function as the new inventory adjustment value of the station and get a new inventory program $S_0$
7: Carry out the number of vehicles used and the corresponding route planning on the basis of the new inventory program $S_0$ and acquire a new route scheme $X^*$
8: Calculate total costs respectively
9: **if** $c(X^*) < c(X_{FC})$ **then**
10: $X_{FC} = X^*$, $FC = f(X^*)$, $c(X_{FC}) = c(X^*)$
11: **end if**
12: **until** the maximal number of iterations without improvement is achieved
13: **Output** the final optimal solutions
4. Numerical study

In the remaining part of this section, we present the performance of the proposed method compared to the original scheme solving the problem independently. Then we provide the sensitivity analyses on key parameters, such as the maximal capacity of a vehicle, unit penalty costs for customer dissatisfaction events, unit inventory holding cost and the observation period of rebalancing. We use MATLAB programming language to write algorithms on a computer configured with an Intel (R) Core (TM) i5-8250U, 1.80 GHz, 8.00 G RAM, running the Windows 10 operating system.

4.1. Experimental data

Our research is based on the docked bike-sharing system and choose the bike-sharing trip data supplied by Capital Bikeshare in Washington for a numerical study. Schuijbroeka et al. [14] and Haider et al. [15] also used the data from Capital Bikeshare to study. Capital Bikeshare, a company which provides bike-sharing service, publishes bike-share trip data in the Washington D.C. and packages it up on its website each quarter. The data includes the time, duration, starting and ending stations for each trip as well as the latitude and longitude of all stations. We use the the latitude and longitude of stations to calculate distance matrix. The latitude and longitude data of each point are further processed and converted into the Euclidean distance between the points. We determine the number of pickups at a station during the observation period using the start date and origin station, as well as the number of returns using the end date and destination station. In this paper, we only considered reallocating shared bikes when the bike-sharing system is idle, such as during nighttime or lunchtime. Therefore, we extracted the mean pick-up and return demand rates of each station for each minute from 6 am to 10 am on a daily basis from the Capital Bikeshare data of the third quarter of 2022 for rebalancing the bike-sharing system at night. To illustrate the feasibility of the proposed formulations, we randomly selected 10, 30, 50, and 70 stations to generate corresponding examples. According to the latitude and longitude data of stations, select the central location as the distribution center.

As a result of limited access to data, the number of broken shared bikes at each station is also randomly generated in [0, 6]. The inventory capacity and the current inventory level at every station are set to 15 and 7 respectively. The observation period is \( t = 5 \) h to ensure the normal operation of the bike-sharing system between 6 am and 10 am. Also, we provided enough homogeneous transport vehicles with a capacity of 15 in the numerical examples. Other parameters are \( c_1 = 50, c_2 = 100, c_3 = 2, c_4 = 2, h = 1, f_k = 50, E_k = 0.8, e_k = 0.02 \).

4.2. Computational results

The goal of this part is to show the benefits of recycling broken bikes within the rebalancing process as opposed to recycling them in isolation. It also aims to highlight the superiority of the integrated inventory routing decision by comparing it to the conventional decentralized decision-making approach (Original) in the context of the bike-sharing system’s inventory routing problem (BIRP).

Table 5 compares the total cost of independent recycling and BIRP solutions in terms of the damage rate, ranging from 10% to 40%. Damage rate refers to the proportion of broken bikes to the total number of bikes. Overall, regardless of changes in station scale or damage rate, recycling bikes during the rebalancing process results in lower total costs compared to individual recovery, with an average improvement rate exceeding 5%. Specifically, under the same station scale, a damage rate of 30% consistently results in the highest improvement rate for the total cost. When the damage rate is low, there are fewer broken bikes, which in turn requires fewer transport vehicles for independent recycling, resulting in reduced total costs. Conversely, when the damage rate is high, there are more broken bikes, and the transport vehicles tend to have higher occupancy rates during individual recovery, leading to cost savings. Therefore, when the damage rate is either very low or very high, recycling broken bikes during the rebalancing process leads to a less significant improvement in total costs.

Table 6 presents a comparison of the performance between the Original and BIRP solutions, focusing on inventory rebalancing costs, procurement and inventory costs, expected accumulated user loss costs, transportation costs, and total costs. The inventory rebalancing costs are equal to the total costs minus the transportation
Table 5. Comparison of performances between the independent recycling scheme and BIRP.

<table>
<thead>
<tr>
<th>No. of stations</th>
<th>Damage rate</th>
<th>Independent recycling</th>
<th>BIRP</th>
<th>Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>10%</td>
<td>265.065</td>
<td>250.072</td>
<td>5.656%</td>
</tr>
<tr>
<td></td>
<td>20%</td>
<td>319.946</td>
<td>303.031</td>
<td>5.287%</td>
</tr>
<tr>
<td></td>
<td>30%</td>
<td>426.368</td>
<td>357.971</td>
<td>16.042%</td>
</tr>
<tr>
<td></td>
<td>40%</td>
<td>481.520</td>
<td>414.784</td>
<td>13.860%</td>
</tr>
<tr>
<td>30</td>
<td>10%</td>
<td>595.109</td>
<td>563.604</td>
<td>5.294%</td>
</tr>
<tr>
<td></td>
<td>20%</td>
<td>767.660</td>
<td>717.879</td>
<td>6.485%</td>
</tr>
<tr>
<td></td>
<td>30%</td>
<td>987.851</td>
<td>878.913</td>
<td>11.028%</td>
</tr>
<tr>
<td></td>
<td>40%</td>
<td>1261.716</td>
<td>1171.908</td>
<td>7.118%</td>
</tr>
<tr>
<td>50</td>
<td>10%</td>
<td>909.366</td>
<td>852.194</td>
<td>6.287%</td>
</tr>
<tr>
<td></td>
<td>20%</td>
<td>1191.477</td>
<td>1125.354</td>
<td>5.550%</td>
</tr>
<tr>
<td></td>
<td>30%</td>
<td>1584.852</td>
<td>1449.856</td>
<td>8.518%</td>
</tr>
<tr>
<td></td>
<td>40%</td>
<td>2027.508</td>
<td>1913.220</td>
<td>5.637%</td>
</tr>
<tr>
<td>70</td>
<td>10%</td>
<td>1216.992</td>
<td>1090.812</td>
<td>2.151%</td>
</tr>
<tr>
<td></td>
<td>20%</td>
<td>1616.567</td>
<td>1556.871</td>
<td>3.693%</td>
</tr>
<tr>
<td></td>
<td>30%</td>
<td>2177.193</td>
<td>2009.823</td>
<td>7.687%</td>
</tr>
<tr>
<td></td>
<td>40%</td>
<td>2792.242</td>
<td>2651.134</td>
<td>5.054%</td>
</tr>
</tbody>
</table>

costs, as well as the purchase and inventory costs plus the expected accumulated user loss costs. Figure 2 depicts the improvement in key performance indicators in the BIRP approach for different station sizes. In each example of scale calculation, the initial number of shared bikes at each station is relatively sufficient. The BIRP approach reduces total costs by reducing purchasing and inventory costs. However, the expected accumulated user loss costs increase, because the sufficient inventory leads to the reduction of parking space at the station, which may increase the number of customer dissatisfaction incidents when users arrive at the station and fail to find space to return the shared bikes. As the expected accumulated user loss costs rise, the overall costs fall. It might be said that the BIRP method sacrifices customer satisfaction in order to find the best cost solution. In the case of 10 and 30 stations, the initial number of shared bikes at each station is sufficient, but few stations need shared bikes. As a result, the reduction of procurement and inventory costs increase the number of bikes to be removed from the stations. After transshipping extra bikes available from one station to another, the final number of shared bikes available increases on transport vehicles proportionally. These bikes available must be transshipped to distribution center alongside the broken shared bikes, and thus transportation costs rise. In the case of 50 and 70 sites, there are relatively more stations with demand for bikes. After the extra bikes available from more stations can be transferred to other stations with demand, the number of remaining shared bikes available eventually decreases and the total transportation volume of transport vehicles shrinks. However, because the transport vehicles also need to recycle the broken shared bikes at each station, the recycling volume and corresponding transport distance do not decrease. As such, the transport costs drop only marginally. Additionally, the reduction ratios of the inventory rebalancing costs, procurement and inventory costs are high and similar. The procurement cost has the largest influence on the inventory rebalancing costs, and the procurement and inventory costs since the unit price of shared bikes is taken into account and the unit price of shared bikes is substantially higher than the unit inventory holding costs and unit penalty costs. When the overall inventory of bikes at the station decreases, procurement costs fall. The inventory rebalancing costs, and procurement and inventory costs drop correspondingly. Nevertheless, the procurement cost is equal to the entire inventory multiplied by the unit price of shared bikes, and changes in inventory are also multiplied by the price of shared bikes, resulting in a relatively high improvement ratio of the relevant costs. As we can see from Figure 2, the average improvement rate of the total cost is about 70%.
Table 6. Comparison of performances between the original scheme and BIRP.

<table>
<thead>
<tr>
<th>No. of stations</th>
<th>Performance</th>
<th>Original</th>
<th>BIRP</th>
<th>Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Inventory rebalancing costs</td>
<td>828.701</td>
<td>127.485</td>
<td>84.616%</td>
</tr>
<tr>
<td></td>
<td>Procurement and inventory costs</td>
<td>827.000</td>
<td>119.000</td>
<td>85.611%</td>
</tr>
<tr>
<td></td>
<td>The expected loss user costs</td>
<td>1.701</td>
<td>8.485</td>
<td>-398.808%</td>
</tr>
<tr>
<td></td>
<td>Transportation costs</td>
<td>73.087</td>
<td>73.792</td>
<td>-0.964%</td>
</tr>
<tr>
<td></td>
<td>Total costs</td>
<td>901.788</td>
<td>201.277</td>
<td>77.680%</td>
</tr>
<tr>
<td>30</td>
<td>Inventory rebalancing costs</td>
<td>828.701</td>
<td>127.485</td>
<td>84.616%</td>
</tr>
<tr>
<td></td>
<td>Procurement and inventory costs</td>
<td>2084.294</td>
<td>282.833</td>
<td>86.430%</td>
</tr>
<tr>
<td></td>
<td>The expected loss user costs</td>
<td>2078.000</td>
<td>260.000</td>
<td>-262.781%</td>
</tr>
<tr>
<td></td>
<td>Transportation costs</td>
<td>6.294</td>
<td>22.833</td>
<td>-398.808%</td>
</tr>
<tr>
<td></td>
<td>Total costs</td>
<td>216.296</td>
<td>218.930</td>
<td>78.190%</td>
</tr>
<tr>
<td>50</td>
<td>Inventory rebalancing costs</td>
<td>2300.590</td>
<td>501.763</td>
<td>78.190%</td>
</tr>
<tr>
<td></td>
<td>Procurement and inventory costs</td>
<td>2723.000</td>
<td>389.000</td>
<td>85.714%</td>
</tr>
<tr>
<td></td>
<td>The expected loss user costs</td>
<td>7.920</td>
<td>46.583</td>
<td>-488.188%</td>
</tr>
<tr>
<td></td>
<td>Transportation costs</td>
<td>334.247</td>
<td>311.810</td>
<td>6.713%</td>
</tr>
<tr>
<td></td>
<td>Total costs</td>
<td>3065.167</td>
<td>747.393</td>
<td>75.617%</td>
</tr>
<tr>
<td>70</td>
<td>Inventory rebalancing costs</td>
<td>1660.834</td>
<td>557.664</td>
<td>66.423%</td>
</tr>
<tr>
<td></td>
<td>Procurement and inventory costs</td>
<td>1651.000</td>
<td>537.000</td>
<td>67.474%</td>
</tr>
<tr>
<td></td>
<td>The expected loss user costs</td>
<td>9.834</td>
<td>20.664</td>
<td>-110.134%</td>
</tr>
<tr>
<td></td>
<td>Transportation costs</td>
<td>462.956</td>
<td>461.931</td>
<td>0.221%</td>
</tr>
<tr>
<td></td>
<td>Total costs</td>
<td>2123.790</td>
<td>1019.596</td>
<td>51.992%</td>
</tr>
</tbody>
</table>

Figure 2. Changes of the key performance measures.

4.3. Sensitivity analysis

A series of sensitivity analyses are based on the key parameters, namely unit penalty costs for customer dissatisfaction events, unit inventory holding costs and the observation period of rebalancing, the maximal capacity of a vehicle. By varying the values of these parameters, we aim to test the corresponding influences on the main performance indicators: inventory rebalancing costs, the expected accumulated user loss costs, transportation costs, and total costs. Specifically, the influences are illustrated through the ratios of change in the indicators computed by \(\frac{\text{Original Results} - \text{BRP Results}}{\text{Original Results}} \times 100\%\).
4.3.1. The unit penalty cost for customer dissatisfaction events

This part examines the impacts of unit penalty costs. We establish three scenarios according to the comparison of the size of unit penalty costs generated respectively by two types of customer dissatisfaction events: the customer fails to return the shared bikes at the stations and the customer fails to rent the shared bike at the stations. Therefore, these three scenarios are specifically expressed by \( c_3 = 2, c_4 = 2; \ c_3 = 3, c_4 = 2; \ c_3 = 2, c_4 = 3 \).

In Figure 3a, the most notable reduction in inventory rebalancing costs is evident when penalty costs are increased for unmet rental demands. Conversely, the smallest improvement in inventory rebalancing costs is observed when penalty costs are raised for unmet return demands. Figure 3d depicts how rules are applied to examples featuring stations of varying scales, mirroring the total costs. This intuitive relationship stems from higher penalty costs for unmet rental demands, necessitating increased inventory at stations. Consequently, when station inventory levels are only half capacity, most stations have relatively sufficient inventory. This leads to a decrease in the number of bikes required per station, resulting in reduced inventory rebalancing costs and increased improvement margins. Conversely, higher penalty costs for unmet return demands require more space allocation at stations, leading to decreased inventory rebalancing costs but reduced improvement levels. However, with 50 stations, some may have a low probability of being empty when \( c_3 = 3, c_4 = 2 \), necessitating increased inventory to meet demand. This results in higher inventory rebalancing costs and decreased improvements.

Figure 3b illustrates that when \( c_3 = 3, c_4 = 2 \), more expected accumulated user loss costs are incurred. This is because compared to the other two categories, the second category tends to allocate more bikes to stations. However, when the costs of inventory rebalancing decrease, the volume of pick-up and delivery at the station, as well as the number of shared bikes, also decrease. This results in a significant increase in predicted losses. Furthermore, altering the penalty costs affects the cost of inventory rebalancing, which in turn influences the number of bikes taken from or placed at the station. Consequently, transportation costs may increase or decrease accordingly. Thus, there is no clear rule regarding how transportation costs change when the three different types of penalty costs are set, as depicted in Figure 3c.

4.3.2. The unit inventory holding cost

We study the impacts of unit inventory holding costs on different scales of stations. Figure 4 illustrates the changes of the four performance indicators when the unit holding costs vary from 1.0 to 4.0.

From Figure 4a, it’s evident that the ratios of change in inventory rebalancing cost are positive. The BIRP approach, which involves transshipping extra shared bikes between stations, effectively reduces inventory holding costs compared to the original scheme. However, the improvement in inventory rebalancing cost is minimal in the cases of 10 and 30 stations. This is because inventory is relatively sufficient when stations are smaller in size, and costs can decrease by reducing inventory levels. Figure 4b shows how expected user loss costs vary with changes in unit inventory holding costs. In scenarios with 10 and 30 stations, as inventory decreases, the occurrences of customer dissatisfaction events increase, consequently leading to higher expected accumulated loss costs for users. However, in the case of 70 stations, where inventory is insufficient, a greater number of bikes are needed to meet demand. As inventory increases to address this shortage, the occurrence of customer dissatisfaction events, such as users arriving at stations unable to return bikes, also rises. This escalation in customer dissatisfaction events leads to an increase in expected user loss costs.

Figure 4c illustrates an increase in the volatility of transportation costs in the case of 50 or 70 stations. However, transportation costs may decrease as unit inventory holding costs increase in other sizes of stations. This phenomenon occurs because there’s a possibility of improving the vehicle loading rate by increasing transport vehicle pick-up and delivery, although constrained by the capacity of transport vehicles. Finally, Figure 4d indicates that the ratio of change in total costs decreases proportionally with an increase in inventory holding costs. This occurs because inventory costs represent a larger proportion of the total costs, and the impact of changes in inventory costs outweighs that of changes in transportation costs. As a result, variations in inventory costs have a more substantial effect on total costs compared to changes in transportation costs.
4.3.3. The observation period of rebalancing

We study the impact of changes in the observation period of rebalancing on performance indicators. In the example, the observation period is 5 h to ensure the normal operation of the bike-sharing system from 6 am to 10 am. In this part, we add average demand rates of 6–8 am and 6–9 am using the same method as the base to get the data. Figure 5 depicts the results of comparing performance indicators with the observation period varying from 3 h to 5 h.

Figure 5a shows that the improvement ratio of inventory rebalancing costs in observation period $t = 4$ is largest. The reduction in inventory rebalancing cost is smallest during the observation period $t = 3$, because there are significant fluctuations in demand throughout the short observation period $t = 3$. Greater inventory rebalancing costs are allocated to meet the total station demand. However, when $t = 4$, there is relatively little fluctuation in demand for individual vehicles, less demand for sites, and less inventory rebalancing costs to be invested.

Figure 5b shows that the observation duration of 4 h results in the expected accumulated user loss costs to increase most dramatically, and the observation period $t = 5$ leads it to increase the least in the case of 10 and 70 stations. This is because the inventory of these two scales is insufficient in the 4 h rebalancing observation period, and then the inventory drops as a result of the small demand, leading to an increase in the expected user loss costs. The demand fluctuation slides and the inventory is reasonably sufficient when the two scales are in the 5 h rebalancing observation period. Therefore, inventory reduction results in a slight increase in the expected accumulated user loss costs. The expected accumulated user loss costs for 30 stations decrease as the observation period increases from 3 to 5, but the opposite is true for 50 stations. In the case of 30 stations, the
inventory is more than adequate, but the parking space is insufficient. Therefore, as the inventory drops and the parking space expands, the expected accumulated user loss costs fall. However, the inventory for 50 stations is extremely limited. Hence, as the inventory falls, the expected accumulated user loss costs rise. It can be seen from Figure 5c that the transportation costs fluctuate, but in the case of 10 stations, the observation period $t = 4$ has the greatest improvement. This is because there are fewer unbalanced stations at this time, and the number of shared bikes needs to be transferred falls. Conversely, in the case of 50 stations, there is a significant increase in transportation costs during the observation period $t = 4$. This may be due to the incompatibility of vehicle capacity with handling demands. Figure 5d shows that the total costs always decrease first and then increase when the observation period changes from 3 to 5.

4.3.4. The maximal capacity of a vehicle

Figure 6 illustrates the performance improvements through employing the BIRP scheme under different maximal load capacity of a vehicle, where the $x$-axis provides the maximal load capacity of a vehicle ranging from 15 to 35, and the $y$-axis indicates the ratios of change in various indicators. To be more realistic, we set the cost associated with the vehicle to change as the vehicle capacity alters (depicted in Tab. 7).

From Figure 6a, the variation of the maximal capacity of a vehicle has a little effect on inventory rebalance costs for the same instance but does have a significant impact on different sized instances. This is due to the fact that inventory rebalancing costs account for a greater proportion of total costs than transportation costs. The BIRP method cannot change the total inventory of the station by altering the maximal capacity of the transport vehicle and indirectly affecting the inventory rebalance costs. Regarding different sizes of stations, intuitively, the larger number of stations, the more inventory rebalancing costs are required to balance the bike-sharing system. Figure 6b indicates that the expected accumulated user loss costs fluctuate significantly
with the increase in the maximal load capacity. The capacity of transport vehicles has increased and the number of bikes available for transfer between stations has grown. Under the premise of constant total inventory, some stations increase inventory, while some stations reduce inventory. Therefore, the inventory level of each station changes, leading to fluctuations in the expected user loss cost. As indicated in Figure 6c, in terms of the examples of different sizes, the ratio of change in transportation costs increases first and then decreases as the maximal capacity of the vehicle grows. However, the peak corresponding to the maximal vehicle capacity of varying scales is different. In the case of 10 stations, the transportation cost is lowest when the maximal loading capacity of the transport vehicle is 30. In the case of 30 stations, the largest percentage change in transportation costs corresponds to a maximal loading capacity of 30 vehicles. However, in the case of 50 and 70 stations, the peak occurs when the maximal loading capacity of the vehicle is 25. As the maximal loading capacity of transportation vehicles increases, the total number of transportation vehicles dispatched by the distribution center decreases and the number of stations visited by each transportation vehicle increases, which makes the transportation cost decrease. However, when the maximum loading capacity of the transport vehicle is too enormous and the quantities of recycling, picking-up, and delivery in the station remain the same, the full load rate of the transport vehicle decreases, making the transport cost increase.

Figure 6d shows that the total costs rise, and then fall slightly. Since the inventory rebalancing costs change slightly, but the transportation costs alter significantly. The total costs change with the transportation costs. Since transportation costs are such a small part of total costs, changes in transportation costs have a minimal impact on total costs.

**Figure 5.** Performance comparison with different observation periods of rebalancing. (a) Inventory rebalancing costs. (b) The expected user loss costs. (c) Transportation costs. (d) Total costs.
Figure 6. Performance comparison with different maximal capacities of a vehicle. (a) Inventory rebalancing costs. (b) The expected user loss costs. (c) Transportation costs. (d) Total costs.

Table 7. Vehicle capacity changes corresponding to cost change settings.

<table>
<thead>
<tr>
<th>Capacity of vehicle</th>
<th>15</th>
<th>20</th>
<th>25</th>
<th>30</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f_k$</td>
<td>50</td>
<td>52</td>
<td>54</td>
<td>56</td>
</tr>
<tr>
<td>$e_k$</td>
<td>0.02</td>
<td>0.04</td>
<td>0.06</td>
<td>0.08</td>
</tr>
<tr>
<td>$E_k$</td>
<td>0.8</td>
<td>0.85</td>
<td>0.9</td>
<td>0.95</td>
</tr>
</tbody>
</table>

5. Conclusions and perspectives

In this paper, we have investigated setting inventory levels and scheduling vehicle handling routes in the bike-sharing system with recycling broken shared bikes and developed a model-based two-stage iterative algorithm that minimizes total costs. The research question had been decomposed into two sub-questions: inventory rebalancing and route planning. Our algorithm is intriguing in the sense that it extracts the solution from the inventory rebalancing and brings it into the route planning, and then uses the results to guide the improvement of the inventory rebalancing, which can be cycled. In the inventory rebalancing, we use the fourth-order Runge–Kutta method to obtain the state transition probability of each station, and then the optimal inventory level of each station is calculated according to the objective function. In the route planning, we have designed the ALNS algorithm for the vehicle routing problem of simultaneous picking-up and delivery. Finally, the empirical data from the Capital Bikeshare have been employed to conduct a series of numerical analyses. The computational results demonstrate the cost variations across the different damage rates and sizes of bike-sharing networks, and the impacts of various key parameters on inventory rebalancing costs, the expected accumulated user loss costs,
transportation costs and total costs. We have summarized the managerial insights obtained from numerical experiments as follows.

Firstly, the BIRP approach has outperformed the independent recycling and original scheme. By applying the proposed BIRP model, the overall performance of shared bikes during recycling and rebalancing process can be greatly improved. Specifically, when comparing BIRP with an independent recycling scheme, total costs show an average improvement of over 5% across various damage rates. When comparing BIRP with original scheme, the total costs can decrease about 70%. And the most obvious improvement lies in inventory rebalancing costs which drop by about 79% on average. This fact indicates that it is beneficial for the bike-sharing operator if the distribution center monitors the inventory of the bike-sharing system and determines a specific bike-sharing recycling and rebalancing plan from a centralized perspective.

Secondly, the total cost improvement is better in the case of small-scale stations. Hence, we suggest that when our method is applied to the large-scale bike-sharing network, we divide the bike-sharing network stations into regions to reduce network scale and make unified decisions for each region, leading to better results.

Thirdly, when determining the route plan, both the loading capacity of transport vehicles and the transport distance from the distribution center to each station should be considered. Hence, when the inventory of a station is insufficient, transshipment from neighboring stations with extra bikes available to stations in need of bikes are served with higher priorities, rather than from distribution centers. Furthermore, after transport vehicles have successfully performed the rebalancing task on their own, recycling broken bikes during the rebalancing process not only increases the loading rate but also raises the operating efficiency of transport vehicles.

Finally, with regard to the different-sized networks, the optimal inventory level setting and transportation planning in terms of various performance measures can be achieved by correspondingly adjusting the values of key parameters, such as unit penalty costs for customer dissatisfaction events, the observation period of rebalancing and the maximal capacity of a vehicle. Specifically, the maximal capacity of transport vehicles has a significant influence on transportation costs. Bike-sharing operators can use transport vehicles with the optimal capacity according to the network size and the volume of picking-up and delivery, reducing transportation costs. Operators can also choose the optimal observation period of rebalancing or lower unit inventory holding costs by improving inventory management to slash total costs.

In conclusion, against the backdrop of the expanding market scale of shared bikes and the advent of the wave of shared bike scrapping, this paper investigates the recycling and rebalancing of shared bikes, examining the target inventory level setting at each station and the route planning of transportation vehicles to adjust recycling at each station, in order to help shared bike operators reduce total costs, improve user satisfaction, and provide a reference for enterprises when formulating inventory levels and transportation plans. The two-stage iterative algorithm solving the inventory routing problem in this paper can also provide references for other operators providing micro mobility services. The research in this paper will not only help the enterprises to operate efficiently and provide better services to urban residents, but also make a contribution to green and sustainable development. This paper presents a preliminary exploration of the static bicycle-sharing system inventory route problem on the basis of stable demand rates, and treats the stations in this paper as independent queuing systems without interactions. Future research should consider the dynamic balance of the bike-sharing system throughout the day and with interactions among stations. Since only the use of homogeneous vehicles to solve the vehicle route planning problem is investigated in this paper, but it is found in this paper that different transport vehicle loading capacities exert different impacts on the shared bicycle network for different-sized stations. Also, the use of heterogeneous vehicles in the vehicle route problem can be considered in future research.

Acknowledgements

The authors would like to thank the editor and the anonymous referees for their helpful comments and suggestions that greatly improved the quality of this paper. This research is supported by the Hunan Key Laboratory of intelligent decision-making technology for emergency management (2020TP1013) and the National Natural Science Foundation of China under Grant Nos. 71771080.
The data that support the findings of this study are openly available in Capital Bikeshare at http://www.capitalbikeshare.com/.

References


Please help to maintain this journal in open access!

This journal is currently published in open access under the Subscribe to Open model (S2O). We are thankful to our subscribers and supporters for making it possible to publish this journal in open access in the current year, free of charge for authors and readers.

Check with your library that it subscribes to the journal, or consider making a personal donation to the S2O programme by contacting subscribers@edpsciences.org.

More information, including a list of supporters and financial transparency reports, is available at https://edpsciences.org/en/subscribe-to-open-s2o.