



# Article Research on Hybrid Scheduling of Shared Bikes Based on MLP-GA Method

Chuanxiang Ren<sup>1</sup>, Hui Xu<sup>1</sup>, Changchang Yin<sup>2,\*</sup>, Liye Zhang<sup>1</sup>, Chunxu Chai<sup>1</sup>, Qiu Meng<sup>3</sup> and Fangfang Fu<sup>1</sup>

- <sup>1</sup> College of Transportation, Shandong University of Science and Technology, Qingdao 266590, China; renchx@sdust.edu.cn (C.R.); 202182160018@sdust.edu.cn (H.X.); zhangly@sdust.edu.cn (L.Z.); chaichunxu2009@sdust.edu.cn (C.C.); 202283160025@sdust.edu.cn (F.F.)
- <sup>2</sup> College of Electrical Engineering and Automation, Shandong University of Science and Technology, Qingdao 266590, China
- <sup>3</sup> College of Electronic and Information Engineering, Shandong University of Science and Technology, Qingdao 266590, China; 202183130043@sdust.edu.cn
- \* Correspondence: ycc2009@sdust.edu.cn

Abstract: Bike-sharing systems with convenience and flexibility have been appearing more and more in cities and become a necessary tool of travel for people. However, the distribution of bikes is highly unbalanced due to the changes in user demand, which leads to the unfavorable situation of "no bikes available" or "too many bikes" at some bike stations. For this reason, this paper proposes a hybrid scheduling method, which combines truck-based scheduling (TBS) and user-based scheduling (UBS). Firstly, a hybrid scheduling model (HBS) combining TBS and UBS is established. Secondly, a method combining multilayer perceptron and genetic algorithm (MLP-GA) is proposed to solve the model. Thirdly, the HBS model is simulated and analyzed by the example. The results show that the MLP-GA method converges, has a faster running time than the genetic algorithm and can obtain solutions with lower total cost and shorter optimal truck path. Further analysis shows that HBS is more implementable in practice and can shorten the optimal truck path and reduce the scheduling total cost while allowing users to use the shared bike in an affordable way, thus realizing the efficient operation of the shared bike system. Finally, a sensitivity analysis of the reward coefficients is performed. This shows that as the reward coefficient increases, the cost of HBS generally shows an increasing trend when the reward coefficient is small, reaches a maximum value when the reward coefficient is 0.6, and decreases slightly thereafter.

Keywords: shared bicycles; hybrid scheduling; MLP-GA method

# 1. Introduction

With the rapid development of the economy, car ownership has gradually increased, resulting in problems such as traffic congestion and environmental pollution. As an effective method to solve these problems, shared transportation schemes with the advantages of efficient car utilization and reducing car ownership and carbon dioxide emission has been widely implemented around the world [1,2]. Among such schemes, the bike-sharing system (BSS) not only makes people's travel more convenient and flexible but also provides a new way to solve the problem of the "last mile" for residents [3]. The BSS first emerged in Europe and was initially managed by public interest organizations set up by the government or in the form of environmental initiatives, which are known as the first generation [4,5]. Since then, as technology developed, the second and third generations of BSSs gradually emerged in the 1990s [6,7], in which parking stakes, deposits, and internet technology were first used. Further, with the rapid development of communication technology, internet technology, positioning technology, and computer technology, a new generation docked BSS, which provides real-time access to all kinds of shared bike information and allows users to rent and return bikes conveniently through handheld devices, has gained popularity around



Citation: Ren, C.; Xu, H.; Yin, C.; Zhang, L.; Chai, C.; Meng, Q.; Fu, F. Research on Hybrid Scheduling of Shared Bikes Based on MLP-GA Method. *Sustainability* **2023**, *15*, 16634. https://doi.org/10.3390/su152416634

Academic Editor: Tamás Bányai

Received: 31 October 2023 Revised: 29 November 2023 Accepted: 4 December 2023 Published: 7 December 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/). the world [8,9]. For example, internet-based BSSs led by OFO have emerged in China [10], and the BSS City Bike in New York introduced an interactive visualization system to enable data-based management [11]. With the increasing popularity of docked shared bike systems, they also face difficulties in management and scheduling [12–14]. For example, the uneven distribution of bikes leads to "no bikes available" or "too many bikes" at bike stations. This inevitably requires bike scheduling to meet changing demands.

Scholars in the field have studied the scheduling of shared bikes, which is divided into two main categories: truck-based scheduling (TBS) and user-based scheduling (UBS). TBS means that the truck arrives at each bike station in turn according to a planned path, replenishes or removes bikes according to the demand of each bike station, and then returns to the fixed parking place. TBS is usually done by building and solving an optimization model to obtain the optimal path for the truck when the demand for each bike station is known. For example, Ho and Szeto [15] established an optimization model after determining the demand of bike stations, which is solved by an iterative heuristic algorithm. Szeto et al. [16] proposed an optimization model to solve the bike scheduling path problem to save the time spent by the truck on the path during bike scheduling and to satisfy the customer demand as much as possible.

UBS encourages users to voluntarily ride bikes at the oversupplied stations to undersupplied stations, so as to complete the shared bike scheduling. UBS is the result of the development and application of internet and big data technologies, such as GPS technology, smartphones, and APPs, in BSS. These technologies allow for real-time access to the location and usage status of bikes and information related to each bike station, thus providing the support for UBS. UBS is mainly achieved through a reward mechanism, that is, by giving out rewards to motivate users to participate in bike scheduling. Users decide whether to participate in UBS according to their situation. For example, Claudio Ruch et al. [17] proposed to use price as a signal to communicate with users and provide monetary rewards for users to change their destination to the more remote bike stations. Wu et al. [18] developed a model with service level as the optimization objective to change users' bike usage behavior and encourage them to help complete the rebalancing of the BSS through monetary rewards. The method was demonstrated through simulation experiments to be able to achieve service level while gaining greater profit.

Compared to TBS, UBS can reduce the operating cost of BSS [17,18]. However, in practice, the complexity of users' travel purpose and time requirements [19,20], as well as the influence of bike station locations and the time-varying nature of the number of shared bikes make it impossible to perform all tasks of bike scheduling with UBS alone [21,22]. At the same time, TBS is already widely used in practice due to its technical feasibility and operational maturity. Under these circumstances, the combination of TBS and UBS offers a new mode for scheduling of shared bikes. This paper establishes a hybrid scheduling model combining TBS and UBS, and in order to be able to effectively utilize the big data of shared bikes, the MLP-GA method is proposed to solve the model. The contributions of this study mainly involve the following three aspects:

(1) A hybrid scheduling model for shared bikes is established, which minimizes the total cost of TBS and UBS as an objective function, including the fixed cost of trucks, transportation, and loading and unloading costs, as well as the user incentive cost.

(2) In order to solve the established model, the MLP-GA method is proposed, in which the fitness of an individual is calculated by MLP network instead of the complicated calculating process in GA.

(3) The model is simulated based on example, and the results show that HBS can reduce the total scheduling cost while allowing users to use the shared bike in an affordable way and is better than TBS. In addition, the MLP-GA method converges faster than GA and is suitable for solving the scheduling problem of shared bikes.

The rest of this paper is presented as follows. In Section 2, the state of the art of research related to shared bike scheduling methods and multilayer perceptron is presented. In Section 3, the problem of the study is described. In Section 4, a hybrid scheduling model

combining TBS and UBS is established. In Section 5, the MLP-GA method for the proposed model is designed. In Section 6, an example for the demonstration of the proposed model is presented and discussed. Finally, the conclusions and suggestions for future work are presented in Section 7.

#### 2. Literature Review

Shared bike scheduling is related to the level of service and user satisfaction and is one of the core contents of BSS operation management. It has received much attention from scholars and some research achievements have been published. For TBS, Chemla et al. [23] established a model of the bike scheduling problem and its relaxation model, and a branchand-cut algorithm was used to solve the model. Gaspero et al. [24] established a truck path model and a step-by-step model of the problem perspective for the bike-sharing system balance problem, and solved it by a neighborhood search algorithm. Raviv et al. [25] developed different bike inventory and routing models based on different assumptions to solve the problem of inaccurate truck routing and bike scheduling quantity. Caggiani and Ottomanelli [26] proposed a microsimulation model that can redetermine the number of bikes at stations and determine the optimal truck route to achieve the objective of minimizing costs. Kloimüllner et al. [27] proposed a model that meets the demand of each bike station before the end of scheduling and avoids the situation where there is no bike at any one bike station. In this case, the model is solved by an improved greedy algorithm. Erdoğan et al. [28] studied the static bike relocation problem with upper and lower bounds on the bike station inventory. Here, an integer programming model with the objective of minimum cost is developed and solved to obtain the optimal bike scheduling path. Dell'Amico et al. [29] proposed four different mixed integer linear programming models to solve the bike-sharing rebalancing problem according to different types of variables and constraints, and applied branch-and-cut algorithms to solve the models. Erdoğan et al. [30] established an opportunity constraint model for solving the static bike rebalancing problem and then solved the model by a heuristic algorithm. Schuijbroek et al. [31] proposed a constrained planning model for solving the bike rebalancing problem, and solved it based on an improved version of the elimination method. Kadri et al. [32] abstracted the bike scheduling problem as a traveler problem and proposed a model with the objective of minimum total waiting time for customers. The model is solved by the branch-and-bound algorithm and proved through experiments. Pal and Zhang [33] established a hybrid integer linear model for the bike balance problem. The model considers the case of scheduling by a single truck, multiple trucks, or multiple visits of the same truck to the same bike station and is solved by an improved neighborhood search algorithm. Bulhõeset et al. [34] proposed an integer programming model to solve the problem of rebalancing bikes at minimal cost. Shi et al. [35] established a vehicle path problem model for bike inventory rebalancing and vehicle routing for the problem of uneven distribution of bikes. They proposed an improved particle swarm optimization algorithm to solve it. Guo et al. [36] developed an optimization model to solve the problem of uneven layout of bike stations and bike distribution, with the objective of minimum operating time and construction cost. The model considered the number of parking spaces and layout factors of bike stations and was simulated based on the BSS in the Nanjing University of Technology campus.

When it comes to UBS, this has also been extensively studied and a series of results achieved. Cepolina and Farina [37] proposed a multi-user-based approach, similar to UBS, to solve the vehicle imbalance problem in a car-sharing system. A simulation model was developed and solved by a simulated annealing algorithm. Aeschbach et al. [38] proposed four different user scheduling control strategies and demonstrated the effectiveness of user scheduling through simulations on London's Barclays cycle hire scheme. Singla et al. [39] proposed a crowdsourcing mechanism considering the minimum regret approach and optimal pricing strategy, which can provide multiple options to incentivize users to complete the repositioning of bikes, and whose feasibility was verified by simulation. Li and Shan [40] established a two-way incentive model to motivate leisure users to

participate in reallocating bikes while encouraging commuters to avoid peak trips to reduce the scheduling pressure on bikes during peak periods. Three different traveler behaviors were simulated considering the relationship between operators and the government. The optimal subsidy method for different regions is also obtained by analyzing the trend of profit variation under different conditions. Haider et al. [41] used an option-based price mechanism to encourage users to pick up and return bikes from nearby bike stations to reduce the number of unbalanced bike stations. The scheme was demonstrated through simulation to reduce the overall operating cost. Luo et al. [42] proposed a new scheduling method to solve the problem of uneven geographical location distribution between bikes and users. The method divides different scheduling areas for users in different situations, and makes users participate in bike scheduling by setting a reward and punishment mechanism. Pakdeewanich et al. [43] studied the use of BSS on a university campus in Thailand, and proposed an incentive approach to solve the problem of uneven bike distribution based on the lessons of urban bike sharing. Cheng et al. [44] designed a user-based bike rebalancing strategy, in which the incentive mechanism is dynamic, and improved the GetThreshold algorithm to solve it. Simulation results demonstrated that the strategy can yield an incentive price that meets users' expectations and avoid budget wastage, thus effectively improving the service level of the BSS. Jin et al. [45] established a UBS model to balance the number of shared bikes, in which return rewards and pickup rewards are set up through a dynamic reward mechanism; the model was proved experimentally to reduce the operation cost.

The above studies on shared bike scheduling have achieved significant results. However, TBS and UBS each have their advantages and disadvantages, and research results from studies that combined them for bike scheduling have gradually appeared. Pfrommer et al. [46] combined the truck-based bike redistribution path problem with real-time price incentives for users, proposed a model for bike scheduling, and used a heuristic algorithm to obtain reasonable bike allocation for different points in time. Reiss and Bogenberger [47] build a model to predict the demand of the BSS, and then provide an optimal strategy combining UBS and TBS according to the imbalance of bike stations. Svenja and Klaus [48] proposed a method of combing the operator-based scheduling strategy with the user-based scheduling strategy, and analyzed the advantages of combining the two, which provided a basis for further research on the application of both strategies. Chiariotti et al. [49] proposed a bike scheduling method combining user incentives and dynamic truck scheduling, which provides incentives to users to make them help reduce the danger of empty or full bike stalls at bike stations. Simulation results confirmed that the method can achieve better service quality with lower operating costs.

In addition, with the increasing scale of shared bike schemes, the requirements for solving the related algorithms are growing more complex. Some scholars have proposed improved algorithms to solve the model of the TBS problem. Papazek et al. [50] proposed an improved iterative greedy heuristic algorithm based on the preferred iterative look ahead technique method and the greedy randomized adaptive search procedure concept. The superiority of the algorithm is demonstrated by using the Austrian Institute of Technology as an example. Angeloudis et al. [51] proposed a repositioning strategy to determine the location of bikes and the regularly visited bike stations, and verified the effectiveness of the strategy. Forma et al. [52] analyzed the main challenges of the BSS, the demand fluctuations of bikes and vacant lock posts at each rental location. A 3-step mathematical planning approach based on a heuristic static relocation problem was used to develop and solve the bike scheduling model. Brinkmann et al. [53] considered the importance of demand at each bike station to conduct scheduling planning, and designed a short-term relocation strategy to solve the problem. Nair and Miller-Hooks [54] designed an equilibrium network model to achieve a balanced allocation of bike stations through rational scheduling routes to improve the utilization of bikes. Caggiani et al. [55] proposed an effective dynamic bike redistribution method, which can predict the number of bike stations and scheduling routes to achieve higher satisfaction with lower bike repositioning costs. Mao et al. [56] proposed a dynamic scheduling method to solve the irregular bike scheduling problem. In addition, bike stations are clustered, and real-time demand is predicted. Admittedly, the above solution algorithms in the literature effectively solve the shared bike scheduling model. However, with the popularization of BSS in the context of communication technology and network connection technology, a large amount of data about BSS emerges, and the above methods cannot be adapted to solve the shared bike scheduling problem when using big data. The development and application of machine learning provides a way to solve the big data problem of shared bike schemes, e.g., a MLP network. MLP networks are widely used in prediction problems as a machine learning method that can process large-scale data and learn data features useful for prediction [57–59].

The above studies on shared bike scheduling have achieved good results, especially proposed strategies for hybrid scheduling. However, there are still gaps in research on combining TBS and UBS for scheduling shared bikes, such as the impact of user reward coefficients in user scheduling on the overall scheduling. Further, how to utilize machine learning methods to achieve effective use of big data from BSS for scheduling also needs further research. Based on this, this paper establishes a hybrid scheduling model of shared bikes with the objective function of minimizing the sum of the costs of TBS and UBS, in which the cost of UBS is calculated by the user reward coefficient. Moreover, a method combining the MLP network and GA, which can deal with the big data of shared bikes, is proposed to solve the hybrid scheduling model of shared bikes.

#### 3. Problem Description

The BSS includes shared bikes, stations, management center, trucks, as well as the corresponding personnel. The management center can obtain the information about the bike, bike station, truck, and user, which are the basis for bike scheduling. Users rent bikes at the origin station through handheld devices such as cell phones, and then return them after riding to the destination station. Due to the changing demand for bikes, there will be some bike stations with oversupplied bikes while yet others have no bikes available. It is necessary to balance the bikes at each station to meet the user demands, which is the bike scheduling problem.

In this paper, a hybrid scheduling combining TBS and UBS for BSS, namely HBS, is studied. A schematic diagram of BSS with HBS is shown in Figure 1. In Figure 1, the BSS includes a management center, a parking place for trucks, and five bike stations: R<sub>1</sub>, R<sub>2</sub>, R<sub>3</sub>, R<sub>4</sub>, R<sub>5</sub>. In addition, in order to better demonstrate the characteristics of HBS, the schematic diagrams of UBS and TBS are also shown in Figure 1.



Figure 1. The schematic diagram of HBS.

In HBS, based on the station demand, scheduling quantity, and user information, the management center develops the optimal truck path and the optimal user scheduling scheme. Then, bikes at each station are balanced by truck and users. At the same time, the information about completion of bike scheduling is monitored by the management center. However, in TBS, the management center only needs to develop the optimal truck path, and in UBS, only the optimal user scheduling scheme needs to be formulated.

#### 4. Mathematical Model

#### 4.1. Model Assumptions and Notation Description

In order to facilitate the modeling, the following assumptions and notation descriptions are provided.

#### 4.1.1. Model Assumptions

Several basic assumptions are as follows:

- 1. The demand at each bike station does not change at start of the scheduling.
- 2. There are enough trucks and users available.
- 3. The user accepts the bike scheduling task and completes it.
- 4. Each UBS scheduling task can only be completed once by one user and cannot be completed repeatedly.

## 4.1.2. Notation Description

The variables are defined as follows:

*i*, *j*, *l*: the index of bike stations, i = 1, ..., N, j = 1, ..., N, l = 1, ..., N, where *N* is the total number of bike stations

*k*: the index of trucks, k = 1, ..., M, where *M* is the total number of trucks

*m*: the index of users, m = 1, ..., U, where U is the total number of users

*n*: the index of bikes, n = 1, ..., B, where *B* is the total number of bikes

 $W_i$ : the scheduling quantity of the bike station *i* 

 $q_{ki}$ : the scheduling task quantity of truck k at bike station i

 $d_{ij}$ : the distance between the bike stations *i* and *j* 

*E*: the capacity of a truck

 $R_{kij}$ : the number of bikes loaded by truck k from bike station i to bike station j

 $x_{kij}$ : decision variable—if truck *k* travels from station *i* to station *j*,  $x_{kij} = 1$ ; otherwise  $x_{kij} = 0$ 

 $\delta_{imn}$ : decision variables—if user *m* participates in the UBS of bike *n* at bike station *i*,  $\delta_{imn} = 1$ ; otherwise,  $\delta_{imn} = 0$ 

 $Z_{mi}$ : the number of bikes parked at station *i* when user *m* participated in UBS

 $F_{mi}$ : the number of bikes ridden from station *i* when user *m* participated in UBS *G*: the set of bike stations

*G*<sub>1</sub>: the set of bike stations in TBS, *G*<sub>1</sub>  $\subseteq$  *G* 

*G*<sub>2</sub>: the set of bike stations in UBS,  $G_2 \subseteq G$ 

### 4.2. Mathematical Model Building

A mathematical model for shared bike scheduling combining TBS and UBS is formulated, and the objective function is to minimize the total costs of TBS cost and UBS cost.

(1) Cost of TBS

The process of TBS requires the coordination of bikes, trucks, users, and operators, during which the operators want to get more revenue while users want convenient and affordable use of bikes. The costs incurred in the process mainly include staff wages, the cost of using the truck, and the cost of loading and unloading bikes. In the paper, the sum of fixed cost, transportation cost, and loading and unloading costs are used to represent the cost of TBS.

7 of 23

#### (1) Fixed cost

The fixed cost involves many aspects, such as truck driver and other staff wages, truck acquisition and maintenance costs, and depreciation cost. After comprehensive consideration of these costs, it is summarized as a fixed cost for each truck, and the calculation formula, labeled as  $Z_1$ , is as follows:

$$Z_1 = C_1 M \tag{1}$$

where  $C_1$  is the fixed cost per truck, in dollars per truck.

Transportation cost

In the process of transporting shared bikes, trucks will consume fuel, which is called transportation cost, labeled as  $Z_2$ . It is usually calculated according to the distance the truck travels, and the calculation formula can be expressed as follows:

$$Z_2 = C_2 \sum_{k \in M} \sum_{i \in G_1} \sum_{j \in G_1} x_{kij} d_{ij}$$
(2)

where  $C_2$  is the truck's transportation cost per kilometer, in dollars per kilometer.

③ Loading and unloading costs

The loading and unloading costs are a portion of the salary paid to the workers for replenishing or removing bikes during the bike scheduling process. The loading and unloading costs are related to the scheduling quantity of the station, which is the number of bikes to be replenished or removed from the station. The calculation formula of loading and unloading costs, labeled as  $Z_3$ , can be expressed as follows:

$$Z_3 = C_3 \sum_{k \in M} \sum_{i \in G_1} |q_{ki}|$$
(3)

where  $C_3$  is the loading and unloading cost per bike, in dollars per bike.

(2) Cost of UBS

In UBS, the reward is used as an incentive to motivate users to participate and complete bike scheduling tasks. The reward is given directly to the user by the bike operator and is the cost of the UBS, which is essentially a portion of the user's rent for the bike. The calculation formula of cost of UBS, labeled as  $Z_4$ , can be obtained as follows:

$$Z_4 = \sum_{i \in G_2} \sum_{m \in U} \sum_{n \in B} \delta_{imn} \sum_{i \in G_2} \sum_{j \in G_2} d_{ij} \times C_4 \times \alpha \tag{4}$$

where  $C_4$  is the rent per meter of the bike and  $\alpha$  is the reward coefficient,  $\alpha \in [0, 1]$ .

(3) The hybrid scheduling model

Based on Formulas (1)–(4), the hybrid scheduling model can be obtained as follows:

$$Min \ Z = C_1 M + C_2 \sum_{k \in M} \sum_{i \in G_1} \sum_{j \in G_1} x_{kij} d_{ij} + C_3 \sum_{k \in M} \sum_{i \in G_1} |q_{ki}| + \sum_{i \in G_2} \sum_{m \in U} \sum_{n \in B} \delta_{imn} \sum_{i \in G_2} \sum_{j \in G_2} d_{ij} \times C_4 \times \alpha$$
(5)

subject to:

$$\sum_{i \in G_1} x_{kij} \le 1, \quad i = 0, k \in M \tag{6}$$

$$\sum_{i \in G_1} x_{kij} \le 1, \quad j = 0, k \in M \tag{7}$$

$$\sum_{i \in G_1} x_{kij} = 1, \quad i \neq j, j \in G_1, k \in M$$
(8)

$$\sum_{j \in G_1} x_{kij} = 1, \quad i \neq j, i \in G_1, k \in M$$
(9)

$$R_{kjl} = R_{kij} + q_{kj}, \quad i \neq j \neq l, i, j, l \in G_1, k \in M$$

$$\tag{10}$$

$$0 \le R_{kij} \le E, \quad i, j \in G_1 \text{ or } i = 0, j \in G_1, i \ne j, k \in M$$
 (11)

$$Z_{mi} + q_{ki} \le W_i, \quad i \in G, k \in M \tag{12}$$

$$F_{mi} + q_{ki} \le W_i, \quad i \in G, k \in M \tag{13}$$

$$G_1 + G_2 = G \tag{14}$$

Formula (5) is the objective function with the minimum total cost. Formulas (6) to (14) are constraints. Constraints (6) and (7) enable the truck to start from the fixed parking place and return to it after completing the bike scheduling tasks. Constraints (8) and (9) ensure that each bike station is served by the truck only once. Constraint (10) ensures the loading quantity of the truck departing the bike station *j* is equal to the sum of the loading quantity of the bike station *i* and the scheduling quantity of the bike station does not exceed the truck capacity and is greater than or equal to 0. Constraints (12) and (13) ensure that the scheduling quantity of bike stations. Constraint (14) guarantees that the sum of bike stations by TBS and UBS in HBS does not exceed the total scheduling quantity of bike stations.

(

#### 5. Algorithm for the Model

#### 5.1. Multilayer Perceptron

An MLP network is a feed-forward neural network that has gained applications in many fields by combining neurons to fit strongly nonlinear data [60,61]. The basic units of an MLP network are neurons, and the weights are optimized during training using a back-propagation algorithm. Each neuron receives a set of input signals and produces an output signal. These input signals are passed to the neuron through connections with weights, which are then processed by an activation function to produce the neuron's output. An MLP network mainly consists of an input layer, a hidden layer, and an output layer [62]. Each layer is fully connected to the next layer to form a forward propagation structure. The input layer receives original data or feature vectors as input and passes them to the hidden layer. The hidden layer is used for feature extraction and representation of the input and can contain multiple neurons, each connected to all the neurons in the previous layer. The output layer produces the final prediction or classification results.

The following is the calculation process from the input to the output of the MLP network with one input layer, *K* hidden layers, and one output layer, as used in the paper. It is assumed that the input and output of the MLP network are *X* and *Y*, respectively; the neuron numbers of input layer and output layer are  $N_I$  and  $N_O$ , respectively; and the neuron number of each hidden layer is  $N_h^i$ , i = 1, 2, 3, ..., K.

The input value of neuron in the first hidden layer is calculated as follows:

$$I_{H_{1i}} = \sum_{j=1}^{N_I} w_{ji1} X \quad i = 1, 2, 3, \dots, N_h^1$$
(15)

where  $H_{1i}$  is the *i*th neuron in the first hidden layer,  $I_{H_{1i}}$  is the input value of neuron  $H_{1i}$ , and  $w_{ji1}$  is the weight between neuron  $H_{1i}$  and the *j*th neuron in the input layer.

Moreover, the output value of neuron in the first hidden layer can be calculated as follows:

$$O_{H_{1i}} = \theta_{H_1} (I_{H_{1i}} + b_{H_1}) \quad i = 1, 2, 3, \dots, N_h^1$$
(16)

where  $o_{H_{1i}}$  is the output value of neuro  $H_{1i}$ ,  $\theta_{H_1}$  is the activation function of the first hidden layer, and  $b_{H_1}$  is the bias of the first hidden layer.

Based on Formula (15), the input value of neuron in the *k*th hidden layer can be calculated as:

$$I_{H_{ki}} = \sum_{j=1}^{N_h^{k-1}} w_{jik} O_{H_{(k-1)j}} \quad k = 2, 3, \dots, K, \ i = 1, 2, 3, \dots, N_h^k$$
(17)

where  $H_{ki}$  is the *i*th neuron in the *k*th hidden layer,  $I_{H_{ki}}$  is the input value of neuron  $H_{ki}$ ,  $w_{jik}$  is the weight between the neuron  $H_{ki}$  and the *j*th neuron in the *k*-1th hidden layer, and  $O_{H_{(k-1)j}}$  is the output value of neuron  $H_{(k-1)j}$ .

Based on Formula (16), the output value of the neuron in the *k*th hidden layer can be calculated as follows:

$$O_{H_{ki}} = \theta_{H_k} (I_{H_{ki}} + b_{H_k}) \quad i = 1, 2, 3, \dots, N_h^k$$
(18)

where  $O_{H_{ki}}$  is the output value of neuron  $H_{ki}$ ,  $\theta_{H_k}$  is the activation function of the *k*th hidden layer, and  $b_{H_k}$  is the bias of the *k*th hidden layer.

Finally, the input value of neuron in the output layer can be calculated as follows:

$$I_{Y_i} = \sum_{j=1}^{N_h^K} w_{ji} O_{H_{K_j}} \quad i = 1, 2, 3, \dots, N_O$$
<sup>(19)</sup>

where  $Y_i$  is the *i*th neuron in the output layer,  $I_{Y_i}$  is the value of neuron  $Y_i$ ,  $w_{ji}$  is the weight between the neuron  $Y_i$  and the *j*th neuron in the *K*th hidden layer, and  $O_{H_{kj}}$  is the output value of neuron  $H_{Kj}$ .

Finally, the output value of the MLP network is calculated as follows:

$$Y = \theta_O (I_Y + b_O) \tag{20}$$

where  $\theta_O$  is the activation function of the output layer,  $I_Y$  is the input value of the output layer which is calculated by Formula (19), and  $b_O$  is the bias of the output layer.

#### 5.2. Genetic Algorithm

A GA is a global optimization algorithm, which continuously retains highly adaptive chromosomes through selection, crossover, and mutation operations to eventually find the optimal solution to the problem [63]. GAs use probabilistic transfer rules, which are operative and can perform parallel searches; they have the advantages of being general and more intelligent, and are very efficient for solving optimization problems. The steps of a GA are as follows.

 Coding Method. Coding refers to the representation of the solution of the problem in terms of individuals consisting of strings. The choice of coding method is related to the characteristics of the problem. The common coding methods are binary encoding and natural number encoding.

(2) Initial Population. The number of individuals in the population is first determined, and then a set of individuals is randomly generated as the initial population based on the chosen coding method.

③ Fitness Function. The fitness function can map each individual to fitness according to the characteristics of the problem and the solution objective, which is used to measure the individual's advantages and disadvantages.

④ Selection Operation. The selection operation is a mimic of the natural selection principle of survival of the fittest, so that individuals with higher fitness have a higher probability to participate in the crossover and mutation operations, thus improving the population generation by generation and optimizing the solution to the problem. In addition, individuals with lower fitness have a chance to be selected to maintain the diversity of the population and thereby avoid falling into local optimal solutions.

10 of 23

(5) Crossover Operation. The crossover operation mimics the phenomenon of gene crossover in biology. In this process, two selected parent individuals produce new offspring individuals by crossover. This combines the good characteristics of the parent individuals thereby increasing the diversity of the population and helping in the search for better solutions in the solution space.

<sup>(6)</sup> Mutation Operation. Mutation operation mimics the process of mutating genes in nature. By randomly changing the genes of individuals, mutation operation is able to introduce new solutions and thus increase the diversity of the population.

⑦ Convergence Criterion. Convergence criterion refers to the conditions under which the genetic algorithm stops iterating. The choice of convergence criterion depends on the characteristics of the problem and the needs to be met by the algorithm. The common convergence criteria are the maximum number of iterations and condition of the fitness value reaching a given threshold.

#### 5.3. MLP-GA

In the current situation where the scale of the shared bike scheme is getting bigger and bigger, the use of GA to solve the model will encounter the problem of the large amount of data for chromosome and population, which requires a lot of time to calculate their related fitness. For this reason, an MLP network that has the ability to handle large data is combined with GA to solve the hybrid scheduling model for shared bikes, and a MLP-GA method is thus obtained. Figure 2 shows the specific flow of the MLP-GA method.



Figure 2. Flowchart of MLP-GA.

In MLP-GA, the MLP network is designed and used to calculate the fitness instead of using the GA's fitness function. In the process, the input of the MLP network is the individual and its fitness, the hidden layer of MLP learns the features related to the individual and its fitness, and the output is fitness. The parameters of the MLP network are then determined, the MLP network is trained by the individual, and its fitness produced by GA, so that the MLP network learns the correspondence between the individual features and the fitness. This enables the MLP to calculate fitness.

The design of the MLP-GA is as follows.

(1) Design of the MLP network. The design of the MLP network includes structure and parameter setting. The structure of the MLP network includes one input layer, two hidden layers, and one output layer. The parameter setting of the MLP network is discussed in Section 6.2.2.

(2) Coding method. The natural number coding method is used. The chromosome consists of the natural numbers, for example, one chromosome "123456", in which the natural numbers correspond to the bike stations.

③ Population initialization. According to the coding method, the population is generated randomly.

④ Genetic operation. The roulette method is used to perform selection operations. The single-point crossover method is used for the crossover operation. The mutation operation is carried out using the method of inverted mutation, in which two mutation points are randomly selected in the parent chromosome, and the probability of mutation is used to determine whether to invert the genes between the two mutation points.

(5) The maximum number of iterations is used as a convergence criterion.

#### 6. Example Simulation and Analysis

# 6.1. Example and Data Processing

To validate the hybrid scheduling model and the MLP-GA method, the BSS of the Rogers Park, Lincoln Park, and their neighborhoods in the north of Chicago, USA, is selected as example. Publicly available BSS data is used, which includes information on the location of shared bike stations and their daily demand, totaling 307,460 items.

In terms of data processing, duplicate data is first removed, and then the missing and abnormal data is processed. For missing data, Kriging interpolation is used for missing longitude or latitude data, and KNN interpolation is used for missing demand data. If two or all of the demand, latitude, and longitude information are missing, then the item is deleted. The data related to usage time of shared bikes not exceeding one minute, or to usage time span being too large are regarded as abnormal, and such data are deleted. After that, the processed data is normalized using the Min–Max method, which reduces the demand, longitude, and latitude values to a uniform range. The final dataset consisting of a total of 257,030 items is obtained.

For this dataset, the distribution of demand for each shared bike station is shown in Figure 3. Considering the reality that users will rent bikes from near stations if there are no bikes available at the origin station, the K-Means method is used for clustering according to the location of shared bike stations. A total of 20 clustering regions are obtained and shown in Figure 4. In Figure 4, the larger red dots represent the center of the clustering region, and the small dots with the other colors represent the shared bike stations. Moreover, the small dots with the same color near the larger red dot belong to the same region; for example, the yellow dots belong to one region and the green dots belong to another region. The 20 clustering region centers are viewed as shared bike stations, which are denoted as  $R_1$ ,  $R_2$ , ...,  $R_{20}$ , respectively, and the shared bike scheduling is developed on this basis. The midday peak hour, AM 12:00 to PM 1:00, is selected for the simulation time. Based on the dataset, the scheduling quantity of the shared bike stations is obtained and shown in Table 1.

# 6.2. Training the MLP Network

#### 6.2.1. Data Preparation

In order to determine the number of neurons in each layer of the MLP network and the parameters such as the activation function and loss function, the MLP network is trained. In the process, GA is used in advance to generate 3000 individuals and their fitness based on the station latitude, longitude, and demand information of the dataset, in which the related parameters are set as follows: the population size is 300, the maximum number of iterations is 1000, the crossover probability is 0.8, and the mutation probability is 0.01.

Next, the first 2100 individuals and their fitness are used as the training set, and the last 900 are used as the test set.



Figure 3. The distribution of demand for each shared bike station.



Figure 4. Clustering results of shared bike stations.

Table 1. Scheduling quantity of bike stations.

| Station         | Scheduling Quantity | Station         | Scheduling Quantity |
|-----------------|---------------------|-----------------|---------------------|
|                 | 11                  | R <sub>11</sub> | 12                  |
| R <sub>2</sub>  | -4                  | R <sub>12</sub> | 7                   |
| R <sub>3</sub>  | 45                  | R <sub>13</sub> | -11                 |
| $R_4$           | -24                 | R <sub>14</sub> | -12                 |
| R <sub>5</sub>  | -13                 | R <sub>15</sub> | 1                   |
| R <sub>6</sub>  | 8                   | R <sub>16</sub> | 10                  |
| R <sub>7</sub>  | -27                 | R <sub>17</sub> | -11                 |
| R <sub>8</sub>  | -14                 | R <sub>18</sub> | 15                  |
| R <sub>9</sub>  | 30                  | R <sub>19</sub> | -10                 |
| R <sub>10</sub> | 1                   | R <sub>20</sub> | -8                  |

6.2.2. Experimental Environment and Parameter Setting

(1) Experimental environment

The hardware and software conditions in the experimental environment are as follows: CPU is Intel i5-8250U @1.60 GHz, RAM is 32 G, the programming language is Python software version 3.7, deep learning framework is TensorFlow version 2.11.

(2) Parameter setting

① The neuron number in the MLP network. The input data of the MLP network is truck path coding, user scheduling scheme coding, and its corresponding fitness; and the number of neurons of the input layer is set as 60. The number of hidden layers is 2, and the number of neurons in them is 128 and 64, respectively. The dimension of the output layer is 1, i.e., the value of the fitness, so the number of neurons is 1.

(2) Loss function. The loss function is used to quantify the difference between the output of the MLP network and the actual target value for a given input dataset. The main goal of the training process is to minimize the value of the loss function and thus improve the performance of the network. The mean square error function is chosen as the loss function for the MLP network.

③ Optimization algorithm. The optimization algorithm minimizes the value of the loss function by iteratively updating the network parameters during training. The Adam algorithm, which can use the first and second-order moment mean of the gradient to calculate the learning rate parameters, is selected as the optimization algorithm for the MLP network.

④ Activation function. The activation function introduces nonlinearity into artificial neural networks, enabling them to learn complex patterns and allowing the network to approximate any arbitrary function. The RULE function is chosen as the activation function for the MLP network.

(5) Other parameters. The batch size is set to 64, the epoch is set as 500 iterations, the regularization technique is L2 regularization, and the evaluation indicator is MAPE.

#### 6.2.3. Analysis of MLP Network Training Results

The MLP network is trained and tested using the designed training and test sets. The loss function curves of the MLP network for the training and test sets are obtained and shown in Figure 5. From Figure 5, the loss function of the training set of the MLP network decreases rapidly and steadily with the increase in the number of iterations and finally converges to a steady state. On the other hand, the loss function of the test set tends to converge to the loss function of the training set after the initial fluctuation and finally leads to a steady state. This suggests that the design of the MLP network is reasonable.



Figure 5. Loss function curves for the training and testing sets of the MLP network.

14 of 23

Figure 6 shows the calculation results of the MLP network in the test set. It can be seen that the fit between the fitness obtained from the MLP network and the actual values is very good. Moreover, the overall error remained stable, and most of them varied within a range of -1 and 1.



Figure 6. Comparison of GA and MLP-GA fitness.

In addition, the MAPE curve was obtained, as shown in Figure 7. In Figure 7, it can be seen that the MAPE curve first experienced a period of fluctuation and then gradually decreased in value and stabilized, and finally converged to 2.67%. This indicates that the MLP network has good robustness and small error, and is well suited to realizing the calculation of individual fitness.



Figure 7. MAPE curve of MLP network for the test set.

Furthermore, the comparison of the convergence curves of MLP-GA with GA for 1000 iterations under the same conditions and parameter settings is shown in Figure 8. The running time of MLP-GA and GA are obtained and shown in Table 2.

As can be seen in Figure 8, the MLP-GA converges after 300 generations while GA does not converge, which exemplifies how the powerful learning and calculating capability of a MLP network simplifies the complex process of calculating the fitness of GA and speeds up the convergence. In addition, it can be seen in Table 2 that the running time of MLP-GA is shorter. This indicates MLP-GA can improve computational efficiency and has better performance than GA.



Figure 8. Convergence speed comparison between MLP-GA and GA.

Table 2. The running time of MLP-GA and GA.

| Method | Time (minute) |
|--------|---------------|
| MLP-GA | 16            |
| GA     | 23            |

#### 6.3. Results and Analysis

#### 6.3.1. Simulation of HBS by MLP-GA

Based on Formula (5), the example is modeled and simulated using Python. In the simulation process, the parameter settings of GA in MLP-GA are the same as in Section 6.2.1. The settings of other related parameters are shown in Table 3, among which the reward coefficient is set based on reference [64].

Table 3. Relevant parameter settings.

| Parameter             | Value and Unit   |
|-----------------------|--|
| Truck capacity        | 60 bikes   |
| <i>C</i> <sub>1</sub> | 30 dollars   |
| <i>C</i> <sub>2</sub> | \$1/km   |
| <i>C</i> <sub>3</sub> | \$0.5/bike   |
| C4                    | There is a \$1 charge for the first 1000 m, and \$0.5 for each additional 500 m. Less than 500 m are counted as 500 m. |
| α                     | 0.4  |

After numerous simulations, the optimal user scheduling scheme for UBS is obtained, as shown in Table 4. The bike scheduling tasks that are not completed by UBS are completed by TBS, and the optimal truck path is obtained, as shown in Table 5. The schematic diagram of the optimal hybrid scheduling scheme is shown in Figure 9. Furthermore, the scheduling quantity and loading quantity of truck at each bike station can be obtained as shown in Figure 10.

From Table 4, it can be seen that in the optimal user scheduling scheme, three users are needed to ride from station  $R_1$  to station  $R_2$ , eight users to ride from station  $R_1$  to station  $R_{20}$ , etc. As a result, a total of 60 users are needed to participate in UBS with a user incentive cost of \$119.20. From Table 5, it can be seen that the truck starts from the fixed parking place, arrives at station  $R_3$ , station  $R_{15}$ , and other stations to complete the scheduling task, and finally returns to the fixed parking place. The length of the optimal truck path is 76.16 km. From Tables 4 and 5, the total cost of HBS is obtained as \$221.36.

| User's Origin Station | User's Destination Station | Number of Bikes Scheduled by Users |
|-----------------------|----------------------------|------------------------------------|
| R <sub>1</sub>        | R <sub>2</sub>             | 3                                  |
| $R_1$                 | R <sub>20</sub>            | 8                                  |
| R <sub>3</sub>        | R <sub>13</sub>            | 9                                  |
| R <sub>3</sub>        | R <sub>17</sub>            | 6                                  |
| $R_9$                 | R <sub>14</sub>            | 12                                 |
| $R_9$                 | R <sub>17</sub>            | 5                                  |
| R <sub>11</sub>       | R <sub>13</sub>            | 2                                  |
| R <sub>11</sub>       | R <sub>19</sub>            | 10                                 |
| R <sub>18</sub>       | R <sub>2</sub>             | 1                                  |
| R <sub>18</sub>       | R <sub>7</sub>             | 4                                  |

Table 4. The optimal user scheduling scheme for HBS as simulated by MLP-GA.

Table 5. The optimal truck path for HBS as simulated by MLP-GA.

| Truck Path  | Length of the Truck Path (Km) | Cost (Dollar) |
|---|-------------------------------|---------------|
| $\begin{array}{c} R_3 \rightarrow R_{15} \rightarrow R_{11} \rightarrow R_{10} \rightarrow R_5 \rightarrow R_9 \rightarrow R_8 \rightarrow \\ R_6 \rightarrow R_{18} \rightarrow R_1 \rightarrow R_4 \rightarrow R_{16} \rightarrow R_{12} \rightarrow R_7 \end{array}$ | 76.16                         | 102.16        |



Figure 9. Schematic diagram of the hybrid scheduling scheme.



Figure 10. Scheduling quantity and loading quantity of truck.

In Figure 10, the horizontal coordinate is the movement order of the truck, and the vertical coordinates are the values of the scheduling quantity and loading quantity. A positive number of scheduling quantity indicates that the bike station has an oversupply of bikes that need to be removed, while a negative number indicates that the bike station needs to be replenished with bikes. As can be seen from Figure 10, the truck arrives at each

bike station in turn to remove or replenish bikes to meet the demand at that bike station. At bike stations that are oversupplied, the truck will remove the excess bikes. For example, at station  $R_3$ , where the scheduling quantity is 30, the truck with an empty load arrives at the station from the parking place, removes the 30 bikes and leaves with a loading quantity of 30. At bike stations that are undersupplied, the truck will replenish bikes. For example, at station  $R_5$ , where the scheduling quantity is -13, the truck arrives with a loading quantity of 33 and leaves with 20, which means 13 bikes are replenished at the station. Moreover, the maximal loading quantity is 60, which satisfied the capacity of the truck.

#### 6.3.2. Simulation of HBS by GA

The example is simulated using GA with the same parameters and conditions. After numerous simulations, the optimal user scheduling scheme and optimal truck path are obtained, as shown in Tables 6 and 7. The schematic diagram of the optimal hybrid scheduling scheme is shown in Figure 11. The scheduling quantity and loading quantity of truck at each bike station are shown in Figure 12.

As can be seen in Table 6, a total of 44 users are needed to participate in UBS with a user incentive cost of \$128.33. From Table 7, the length of the optimal truck path is 80.95 km. From Tables 6 and 7, the cost of HBS is \$244.28. The number of bikes replenished or removed by the truck at each station and the loading quantity can be obtained from Figure 12.

Table 6. The optimal user scheduling scheme for HBS as simulated by GA.

| User's Origin Station | User's Destination Station | Number of Bikes Scheduled by Users |
|-----------------------|----------------------------|------------------------------------|
| R <sub>1</sub>        | R <sub>2</sub>             | 1                                  |
| R <sub>1</sub>        | R <sub>19</sub>            | 2                                  |
| R <sub>1</sub>        | R <sub>20</sub>            | 8                                  |
| R <sub>3</sub>        | R <sub>2</sub>             | 3                                  |
| R9                    | R <sub>13</sub>            | 7                                  |
| R9                    | R <sub>17</sub>            | 11                                 |
| R <sub>11</sub>       | R <sub>13</sub>            | 4                                  |
| R <sub>11</sub>       | R <sub>19</sub>            | 8                                  |

Table 7. The optimal truck path for HBS as simulated by GA.

| Truck Path   | Length of the Truck Path (Km) | Cost (Dollar) |
|--|-------------------------------|---------------|
| $\begin{array}{c} R_{15} {\rightarrow} R_{3} {\rightarrow} R_{5} {\rightarrow} R_{10} {\rightarrow} R_{9} {\rightarrow} R_{11} {\rightarrow} R_{1} {\rightarrow} R_{8} {\rightarrow} \\ R_{14} {\rightarrow} R_{18} {\rightarrow} R_{4} {\rightarrow} R_{12} {\rightarrow} R_{6} {\rightarrow} R_{16} {\rightarrow} R_{7} \end{array}$ | 80.95                         | 115.95        |



Figure 11. Schematic diagram of the hybrid scheduling scheme.



Figure 12. Scheduling quantity and loading quantity of truck.

From Figures 9 and 11, although the number of scheduling tasks of users in the optimal user scheduling scheme obtained by the MLP-GA method is greater, the distance between the original station and the destination station in the scheduling scheme is smaller, so the user's riding cost is less, and the resultant UBS cost is also lower. Comparing Tables 5 and 7, the optimal truck path obtained by the MLP-GA method is shorter and the TBS cost is lower. Therefore, the HBS obtained by the MLP-GA method has a lower cost. This shows that the MLP-GA method performs better than GA and can be used to solve the shared bike scheduling optimization problem.

# 6.3.3. Simulation of TBS by MLP-GA

In the HBS model (Equation (5)), the HBS model becomes the TBS model without considering the user participation problem. The model is simulated using the MLP-GA method for the same example, as well as using the same parameters and software and hardware conditions. After the simulation, the optimal truck path and cost of TBS are obtained, as shown in Table 8. Meanwhile, the scheduling quantity and loading quantity of truck at each bike station are obtained, as shown in Figure 13.



Table 8. Simulation results.

Figure 13. Scheduling quantity and loading quantity of truck.

It can be found from Tables 5 and 8 that scheduling costs increase by \$31.47 when using only TBS compared to HBS. The path increased by 9.67 km. In addition, the scheduling scheme of HBS is conducive to the formation of good interaction between users and operators and the sustainable development of the shared bike system.

#### 6.3.4. Simulation of UBS by MLP-GA

In the HBS model (Equation (5)), the HBS model becomes the UBS model without considering TBS. The model is simulated using the MLP-GA method for the same example, as well as using the same parameters and software and hardware conditions. After the simulation, the optimal user scheduling scheme is obtained, as shown in Table 9.

| User's Origin Station | User's Destination Station | Number of Bikes Scheduled by Users |
|-----------------------|----------------------------|------------------------------------|
| R <sub>1</sub>        | R <sub>8</sub>             | 11                                 |
| R <sub>3</sub>        | $R_4$                      | 19                                 |
| $R_3$                 | R <sub>13</sub>            | 9                                  |
| R <sub>3</sub>        | R <sub>17</sub>            | 11                                 |
| R <sub>6</sub>        | R <sub>14</sub>            | 8                                  |
| $R_9$                 | R <sub>2</sub>             | 4                                  |
| R9                    | $R_5$                      | 13                                 |
| R <sub>9</sub>        | $R_8$                      | 3                                  |
| $R_9$                 | R <sub>14</sub>            | 4                                  |
| R <sub>9</sub>        | R <sub>20</sub>            | 6                                  |
| R <sub>10</sub>       | R <sub>19</sub>            | 1                                  |
| R <sub>11</sub>       | R <sub>13</sub>            | 2                                  |
| R <sub>11</sub>       | R <sub>19</sub>            | 9                                  |
| R <sub>11</sub>       | R <sub>20</sub>            | 1                                  |
| R <sub>12</sub>       | $R_7$                      | 7                                  |
| R <sub>15</sub>       | R <sub>20</sub>            | 1                                  |
| R <sub>16</sub>       | $R_4$                      | 5                                  |
| R <sub>16</sub>       | R <sub>7</sub>             | 5                                  |
| R <sub>18</sub>       | $R_7$                      | 15                                 |

Table 9. The optimal user scheduling scheme for UBS as simulated by MLP-GA.

As can be seen in Table 9, a total of 134 users are needed to participate in UBS with a user incentive cost of \$198.67. From Tables 8 and 9, the cost of UBS is \$54.16 less than the cost of TBS. At the same time, UBS also facilitates the use of bikes by users. However, considering the limitation of users' traveling time during peak hours, it is difficult to complete the scheduling only by UBS at this time, so HBS has a wider range of application situations.

#### 6.4. Sensitivity Analysis

In order to further explore the effect of the reward coefficient on hybrid scheduling, the reward coefficient was selected for sensitivity analysis. The reward coefficient was set as 0.1 to 0.9 with an interval of 0.1, and simulated. The costs of UBS, TBS, and HBS for different reward coefficients were obtained, as shown in Figure 14.

Based on Figure 14, the following conclusions can be drawn:

(1) As the reward coefficient increases, the cost of HBS generally shows an increasing trend when  $\alpha$  is small, reaches a maximum value when the reward coefficient is 0.6, and decreases slightly thereafter. In this case, the cost of UBS increases and then decreases, and the cost of TBS shows a general increasing trend.

(2) The cost of HBS is lowest when the incentive coefficient  $\alpha$  is 0.1. This is due to the fact that the reward coefficient is small and most of the scheduling tasks are done by UBS, which results in a low cost of HBS. In this case, HBS has the lowest cost, but the reward coefficient is so small that in practice it is difficult for any user to agree to participate in UBS. Further, both TBS and UBS costs gradually increase as the incentive factor  $\alpha$  gradually increases from 0.1. This is due to the fact that the increase in the reward coefficient makes

the UBS increase, and the TBS scheduling tasks gradually increase in order to reduce the HBS cost, but at this time, the reward coefficient is at a low level, so the number of HBS scheduling tasks is still high. When the reward coefficient increases to 0.5, the cost of UBS reaches its maximum value and begins to decrease thereafter.

(3) The cost of HBS reaches its maximum when the reward coefficient increases to 0.6, where TBS and UBS are roughly equal. After the reward coefficient exceeds 0.6, the change in HBS costs flattens out, suggesting that there is little difference in scheduling schemes thereafter. At this stage, the reward factor is at a high value, and in order to satisfy the objective of minimizing the total cost, the scheduling scheme starts to decrease the number of UBS, and more scheduling tasks are done by truck, so that the cost of TBS is higher than that of UBS. In this case, although the users are able to use the shared bike in a more affordable way, it is not conducive to the objective of minimizing the operation cost.

(4) Based on the above analysis, the costs of HBS, UBS, and TBS all change with the reward coefficient which can be varied in practice to accommodate changes in conditions. If there is a shortage of trucks, operators can address the imbalance in shared bikes by increasing the reward coefficient to enable more users to participate in UBS. During the peak period of user travel, considering the time urgency of user travel, most of the scheduling tasks will be done by trucks at this time, so the reward coefficient can be appropriately decreased.



Figure 14. The costs of UBS, TBS, and HBS for different reward coefficients.

#### 7. Conclusions and Future Work

To address the imbalance between the supply and demand of bike stations of BSS, a HBS model combing TBS and UBS is established, which takes the minimum total cost of TBS and UBS as the objective function. In the HBS model, TBS cost is expressed as the sum of fixed cost, transportation cost, and loading and unloading cost, while UBS cost is calculated through the rent of users. A method combining multilayer perceptron and genetic algorithm (MLP-GA) is proposed to solve the model.

After that, the mathematical model is simulated by a real example and the results are obtained. In the simulation process, the MLP network is first trained to determine the network parameters and to enable it to calculate the fitness more accurately. It is shown that the MLP-GA is better than GA in terms of convergence speed and running time. Moreover, comparing the scheduling schemes, it can be seen that the MLP-GA method is able to obtain a scheduling scheme with lower total cost and shorter optimal truck path. Further analyses show that HBS is more implementable in practice, enabling the scheduling of shared bikes at a lower cost and allowing users to use shared bikes in an affordable way. In addition, a sensitivity analysis of the reward coefficients is performed. This shows that as the reward coefficient increases, the cost of HBS generally shows an increasing trend when the reward coefficient is small, reaches a maximum value when the reward coefficient is 0.6, and decreases slightly thereafter.

This paper studies the problem of hybrid truck and user scheduling for shared bikes. However, there are some other factors worth considering for optimization, such as the loading rate of truck during scheduling, which need to be taken into account in further research. In addition, although the peak period of bike demand is studied in the paper, it is also worth further research as to whether the same pattern exists for temporary and unexpected supply and demand imbalance problems.

**Author Contributions:** Conceptualization, C.C. and F.F.; Formal analysis, Q.M.; Writing—original draft, H.X.; Writing—review & editing, C.Y.; Supervision, L.Z.; Project administration, C.R. All authors have read and agreed to the published version of the manuscript.

**Funding:** This study was funded by National Natural Science Foundation of China (No. 72101033 and No. 71831001) and Key Research and Development Project of Shandong Province (No. 2019GGX101008).

**Data Availability Statement:** The data source for this paper is: https://www.kaggle.com/datasets/ nabajithazowary95/bikeshare-data (accessed on 1 April 2022).

Conflicts of Interest: The authors declare no conflict of interest.

#### References

- 1. Seik, F.T. Vehicle ownership restraints and car sharing in Singapore. Habitat Int. 2000, 24, 75–90. [CrossRef]
- Nijland, H.; van Meerkerk, J. Mobility and environmental impacts of car sharing in the Netherlands. *Environ. Innov. Soc. Transit.* 2017, 23, 84–91. [CrossRef]
- Leister, E.H.; Vairo, N.; Sims, D.; Bopp, M. Understanding bike share reach, use, access and function: An exploratory study. Sustain. Cities Soc. 2018, 43, 191–196. [CrossRef]
- 4. DeMaio, P. Bike-sharing: History, Impacts, Models of Provision, and Future. J. Public Transp. 2009, 12, 41–56. [CrossRef]
- Shaheen, S.A.; Guzman, S.; Zhang, H. Bikesharing in Europe, the Americas, and Asia: Past, present, and future. *Transp. Res. Rec.* 2010, 2143, 159–167. [CrossRef]
- 6. Shaheen, S.; Guzman, S. Worldwide bikesharing. Access Mag. 2011, 39, 22–28.
- Chen, F.; Turoń, K.; Kłos, M.; Czech, P.; Pamuła, W.; Sierpiński, G. Fifth generation of bike-sharing systems—Examples of Poland and China. Sci. J. Silesian Univ. Technol. Ser. Transp. 2018, 99, 5–13. [CrossRef]
- Mátrai, T.; Tóth, J. Comparative Assessment of Public Bike Sharing Systems. *Transp. Res. Procedia* 2016, 14, 2344–2351. [CrossRef]
   Chun, K.W.; Chen, Z.H.; Sun, C.C. IOT Solution for Public Bicycle System. In Proceedings of the 2016 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW), Nantou, Taiwan, 27–29 May 2016; pp. 47–48.
- Cao, X.Z.; Zhang, J.W.; Ye, C.Q. The development of China's bike-sharing business model—A case study about OFO. In Proceedings of the 4th International Symposium—Management, Innovation and Development, Beijing, China, 25–26 November 2017; pp. 472–479.
- 11. Oliveira, G.N.; Sotomayor, J.L.; Torchelsen, R.P.; Silva, C.T.; Comba, J.L. Visual analysis of bike-sharing systems. *Comput. Graph.* **2016**, *60*, 119–129. [CrossRef]
- 12. Zhang, J.; Pan, X.; Li, M.; Yu, P.S. Bicycle-sharing system analysis and trip prediction. In Proceedings of the 2016 17th IEEE In-ternational Conference on Mobile Data Management (MDM), Porto, Portugal, 13–16 June 2016.
- 13. Kellstedt, D.; Spengler, J.O.; Bradley, K.; Maddock, J.E. Evaluation of free-floating bike-share on a university campus using a multi-method approach. *Prev. Med. Rep.* **2019**, *16*, 100981. [CrossRef]
- 14. Grimes, A.; Baker, M. The effects of a citywide bike share system on active transportation among college students: A ran-domized controlled pilot study. *Health Educ. Behav.* **2020**, *47*, 412–418. [CrossRef]
- 15. Ho, S.C.; Szeto, W. Solving a static repositioning problem in bike-sharing systems using iterated tabu search. *Transp. Res. Part E Logist. Transp. Rev.* **2014**, *69*, 180–198. [CrossRef]
- Szeto, W.Y.; Liu, Y.; Ho, S.C. Chemical reaction optimization for solving a static bike repositioning problem. *Transp. Res. Part D Transp. Environ.* 2016, 47, 104–135. [CrossRef]
- 17. Ruch, C.; Warrington, J.; Morari, M. Rule-based price control for bike sharing systems. In Proceedings of the 2014 European Control Conference (ECC), Strasbourg, France, 24–27 June 2014; pp. 708–713.
- 18. Wu, R.; Liu, S.; Shi, Z. Customer Incentive Rebalancing Plan in Free-Float Bike-Sharing System with Limited Information. *Sustainability* **2019**, *11*, 3088. [CrossRef]
- 19. Ashley, J. Bike sharing as alternative transportation at Bridgewater State University. Undergrad. Rev. 2012, 8, 16–25.
- Yu, Y.; Yi, W.; Feng, Y.; Liu, J. Understanding the Intention to Use Commercial Bike-sharing Systems: An Integration of TAM and TPB. In Proceedings of the 51st Annual Hawaii International Conference on System Sciences (HICSS), Hilton Waikoloa Village, HI, USA, 3–6 January 2018; pp. 646–655.
- Mete, S.; Cil, Z.A.; Özceylan, E. Location and coverage analysis of bike-sharing stations in university campus. *Bus. Syst. Res. J.* 2018, 9, 80–95. [CrossRef]

- 22. Chen, J. Understanding University Student's Decision to Use OFO Bike Sharing Services in Bangkok. 2018. Available online: https://oa.mg/work/3189493936 (accessed on 20 October 2023).
- 23. Chemla, D.; Meunier, F.; Calvo, R.W. Bike sharing systems: Solving the static rebalancing problem. *Discret. Optim.* 2013, 10, 120–146. [CrossRef]
- 24. Di Gaspero, L.; Rendl, A.; Urli, T. Constraint-Based Approaches for Balancing Bike Sharing Systems. In *International Conference on Principles and Practice of Constraint Programming*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 758–773.
- Raviv, T.; Tzur, M.; Forma, I.A. Static repositioning in a bike-sharing system: Models and solution approaches. *EURO J. Transp.* Logist. 2013, 2, 187–229. [CrossRef]
- Caggiani, L.; Ottomanelli, M. A Dynamic Simulation based Model for Optimal Fleet Repositioning in Bike-sharing Systems. Procedia Soc. Behav. Sci. 2013, 87, 203–210. [CrossRef]
- Kloimüllner, C.; Papazek, P.; Hu, B.; Raidl, G.R. Balancing Bicycle Sharing Systems: An Approach for the Dynamic Case. In *European Conference on Evolutionary Computation in Combinatorial Optimization*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 73–84.
- 28. Erdoğan, G.; Laporte, G.; Calvo, R.W. The static bicycle relocation problem with demand intervals. *Eur. J. Oper. Res.* **2014**, 238, 451–457. [CrossRef]
- 29. Dell, M.; Hadjicostantinou, E.; Iori, M.; Novellani, S. The bike sharing rebalancing problem: Mathematical formulations and benchmark instances. *Omega* **2014**, *45*, 7–19. [CrossRef]
- 30. Erdoğan, G.; Battarra, M.; Calvo, R.W. An exact algorithm for the static rebalancing problem arising in bicycle sharing systems. *Eur. J. Oper. Res.* **2015**, 245, 667–679. [CrossRef]
- 31. Schuijbroek, J.; Hampshire, R.; van Hoeve, W.-J. Inventory rebalancing and vehicle routing in bike sharing systems. *Eur. J. Oper. Res.* **2016**, 257, 992–1004. [CrossRef]
- Kadri, A.A.; Kacem, I.; Labadi, K. A memetic algorithm for solving the multiple vehicles routing problem in bicycle sharing systems. In Proceedings of the 12th International Conference on Fuzzy Logic and Intelligent Technologies in Nuclear Science (FLINS), Roubaix, France, 24–26 August 2016; Volume 10, pp. 764–769.
- Pal, A.; Zhang, Y. Free-floating bike sharing: Solving real-life large-scale static rebalancing problems. *Transp. Res. Part C Emerg. Technol.* 2017, 80, 92–116. [CrossRef]
- 34. Bulhões, T.; Subramanian, A.; Erdoğan, G.; Laporte, G. The static bike relocation problem with multiple vehicles and vis-its. *Eur. J. Oper. Res.* **2018**, *264*, 508–523. [CrossRef]
- 35. Shi, L.; Zhang, Y.; Rui, W.; Yang, X. Study on the Bike-sharing Inventory Rebalancing and Vehicle Routing for Bike-sharing System. *Transp. Res. Procedia* 2019, *39*, 624–633. [CrossRef]
- 36. Guo, T.; Yang, J.; He, L.; Tang, K. Emerging technologies and methods in shared mobility systems layout optimization of campus bike-sharing parking spots. *J. Adv. Transp.* **2020**, 2020, 8894119. [CrossRef]
- Cepolina, E.M.; Farina, A. A new shared vehicle system for urban areas. *Transp. Res. Part C Emerg. Technol-Ogies* 2012, 21, 230–243. [CrossRef]
- Aeschbach, P.; Zhang, X.; Georghiou, A.; Lygeros, J. Balancing bike sharing systems through customer cooperation—A case study on London's Barclays Cycle Hire. In Proceedings of the 54th IEEE Conference on Decision and Control (CDC), Osaka, Japan, 15–18 December 2015; pp. 4722–4727.
- Singla, A.; Santoni, M.; Ga 'bor Barto 'k, P.; Mukerji, M.M.; Krause, A. Incentivizing users for balancing bike sharing systems. In Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, Austin, TX, USA, 20–25 January 2015; pp. 723–729.
- 40. Li, L.; Shan, M. Bidirectional Incentive Model for Bicycle Redistribution of a Bicycle Sharing System during Rush Hour. *Sustainability* **2016**, *8*, 1299. [CrossRef]
- 41. Haider, Z.; Nikolaev, A.; Kang, J.E.; Kwon, C. Inventory rebalancing through pricing in public bike sharing systems. *Eur. J. Oper. Res.* **2018**, 270, 103–117. [CrossRef]
- 42. Luo, Y.; Dou, W.; Yan, H.; Liu, L.; Lu, S. An automated planning and scheduling method of shared bikes based on reward and punishment mechanism. In Proceedings of the 2018 17th International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES), Wuxi, China, 19 October 2018.
- Pakdeewanich, C.; Tiyarattanachai, R.; Anantavrasilp, I. Locally Designed Campus Smart Bike Sharing System: Lessons Learned and Design Optimization for Thailand. In Proceedings of the 2020 IEEE 7th International Conference on Industrial Engineering and Applications (ICIEA), Bangkok, Thailand, 18 April 2020; pp. 721–725.
- 44. Cheng, Y.; Wang, J.; Wang, Y. A user-based bike rebalancing strategy for free-floating bike sharing systems: A bidding model. *Transp. Res. Part E Logist. Transp. Rev.* **2021**, 154, 102438. [CrossRef]
- Jin, H.; Liu, S.; So, K.C.; Wang, K. Dynamic incentive schemes for managing dockless bike-sharing systems. *Transp. Res. Part C Emerg. Technol.* 2022, 136, 103527. [CrossRef]
- 46. Pfrommer, J.; Warrington, J.; Schildbach, G.; Morari, M. Dynamic Vehicle Redistribution and Online Price Incentives in Shared Mobility Systems. *IEEE Trans. Intell. Transp. Syst.* **2014**, *15*, 1567–1578. [CrossRef]
- Reiss, S.; Bogenberger, K. Optimal bike fleet management by smart relocation methods: Combining an operator-based with an user-based relocation strategy. In Proceedings of the 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), Rio de Janeiro, Brazil, 1–4 November 2016; pp. 2613–2618.

- 48. Svenja, R.; Klaus, B. A relocation strategy for Munich's bike sharing system combining an operator-based and a user-based scheme. *Transp. Res. Procedia* **2017**, *22*, 105–114.
- 49. Chiariotti, F.; Pielli, C.; Zanella, A.; Zorzi, M. A Bike-sharing Optimization Framework Combining Dynamic Rebalancing and User Incentives. *ACM Trans. Auton. Adapt. Syst.* **2019**, *14*, 1–30. [CrossRef]
- Papazek, P.; Raidl, G.R.; Rainer-Harbach, M.; Hu, B. A PILOT/VND/GRASP hybrid for the static balancing of public bi-cycle sharing systems. In *International Conference on Computer Aided Systems Theory*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 372–379.
- 51. Angeloudis, P.; Hu, J.; Bell, M.G.H. A strategic repositioning algorithm for bicycle-sharing schemes. *Transp. A Transp. Sci.* 2014, 10, 759–774. [CrossRef]
- 52. Forma, I.A.; Raviv, T.; Tzur, M. A 3-step math heuristic for the static repositioning problem in bike-sharing systems. *Transp. Res. Part B Methodol.* **2015**, *71*, 230–247. [CrossRef]
- 53. Brinkmann, J.; Ulmer, M.W.; Mattfeld, D.C. Short-term strategies for stochastic inventory routing in bike sharing systems. *Transp. Res. Procedia* 2015, *10*, 364–373. [CrossRef]
- Nair, R.; Miller-Hooks, E. Equilibrium design of bicycle sharing systems: The case of Washington D.C. *EURO J. Transp. Logist.* 2016, 5, 321–344. [CrossRef]
- 55. Caggiani, L.; Camporeale, R.; Ottomanelli, M.; Szeto, W.Y. A modeling framework for the dynamic management of free-floating bike-sharing systems. *Transp. Res. Part C Emerg. Technol.* **2018**, *87*, 159–182. [CrossRef]
- Mao, D.; Hao, Z.; Wang, Y.; Fu, S. A Novel Dynamic Dispatching Method for Bicycle-Sharing System. *ISPRS Int. J. Geo-Information* 2019, *8*, 117. [CrossRef]
- Ahmadlou, M.; Al-Fugara, A.; Al-Shabeeb, A.R.; Arora, A.; Al-Adamat, R.; Pham, Q.B.; Al-Ansari, N.; Linh, N.T.T.; Sajedi, H. Flood susceptibility mapping and assessment using a novel deep learning model combining multilayer perceptron and autoencoder neural networks. J. Flood Risk Manag. 2020, 14, 12683. [CrossRef]
- 58. Temel, F.A.; Yolcu, C.; Kuleyin, A. A multilayer perceptron-based prediction of ammonium adsorption on zeolite from landfill leachate: Batch and column studies. *J. Hazard. Mater.* **2021**, *410*, 124670. [CrossRef] [PubMed]
- Cheng, Z.; Zhang, Y.; Tang, C. Solving Monocular Sensors Depth Prediction Using MLP-Based Architecture and Multi-Scale Inverse Attention. *IEEE Sens. J.* 2022, 22, 16178–16189. [CrossRef]
- 60. Kasgari, A.A.; Divsalar, M.; Javid, M.R.; Ebrahimian, S.J. Prediction of bankruptcy Iranian corporations through artificial neural network and Probit-based analyses. *Neural Comput. Appl.* **2012**, *23*, 927–936. [CrossRef]
- Moon, T.; Hong, S.; Choi, H.Y.; Jung, D.H.; Chang, S.H.; Son, J.E. Interpolation of greenhouse environment data using multilayer perceptron. *Comput. Electron. Agric.* 2019, 166, 105023. [CrossRef]
- 62. Xiang, C.; Ding, S.; Lee, T.H. Geometrical Interpretation and Architecture Selection of MLP. *IEEE Trans. Neural Netw.* 2005, 16, 84–96. [CrossRef]
- 63. Mirjalili, S. Genetic algorithm. In *Evolutionary Algorithms and Neural Networks;* Studies in Computational Intelligence; Springer: Berlin/Heidelberg, Germany, 2019; pp. 43–55.
- 64. Wang, D.; Liao, F. Incentivized user-based relocation strategies for moderating supply–demand dynamics in one-way car-sharing services. *Transp. Res. Part E Logist. Transp. Rev.* 2023, 171, 103017. [CrossRef]

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.