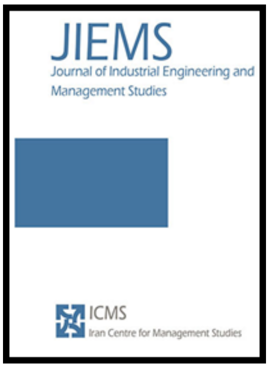
JIEMS
Journal of Industrial Engineering and Management Studies

Vol. 7, No. 1, 2020, pp. 19-34

DOI: 10.22116/JIEMS.2020.110002

www.jiems.icms.ac.ir



Balancing public bicycle sharing system by defining response rates for destinations

Behzad Maleki Vishkaei[1,*], Iraj Mahdavi[1], Nezam Mahdavi-Amiri[2], Esmail Khorram[3]

Abstract

Public Bicycle Sharing System (PBSS) is used as a way to reduce traffic and pollution in cities. Its performance is related to availability of bicycles for picking up and free docks to return them. Existence of different demand types leads to the emergence of imbalanced stations. Here, we try to balance inventory of stations via defining maximal response rates for each type of rental request. If the maximal response rate for a destination is lower than 100 percent, a part of the proposed destination requests is rejected in the hope of balancing the inventory. The goal is to minimize the mean extra inventory and the mean rejected requests by providing proper amounts of the maximal response rates. An approximation method named as Mean Value Analysis (MVA) is used to develop a genetic algorithm for solving the problem. Different examples are worked through to examine the applicability of the proposed method. The results show that the proposed policy leads to a significant improvement and reduces the users' dissatisfaction.

Keywords: Public Bicycle Sharing System (PBSS); Jackson network; Mean Value Analysis (MVA); Maximal response rate.

Received: March 2019-14

Revised: September 2019-24

Accepted: October 2019-12

1. Introduction

In recent years, the interest in Public Bicycle Sharing Systems (PBSS) has increased extensively due to environmental concerns such as pollution and traffic problems. These systems can help us to develop new transit modes for urban areas. Yang et al. (2018) used the data corresponding to the public bicycle-sharing systems of Hangzhou and Ningbo in China to study how the public bicycle-sharing systems affect the original urban public transport networks and showed they could help to reduce the passengers' trip times. Many researchers discussed PBSS problems from different viewpoints.

* Corresponding author; b.maleki.v@gmail.com
[1] Department of Industrial engineering, Mazandaran University of Science and Technology, Babol, Iran.
[2] Department of Mathematical Sciences, Sharif University of Technology, Tehran, Iran.
[3] Department of Mathematics, Amirkabir University of Technology, Tehran, Iran.

Some of them studied the location of stations and the system fleet size. Lin and Yang (2011) studied a model to compute the number of bike stations and their locations. They analyzed different parameters of the model to arrive at a more appropriate design of the system. George and Xia (2011) computed the fleet size with the goal of maximizing the profit of the system. They studied their program as a queuing network model and proposed an approximation method for large problems. Martinez et al. (2012) surveyed the bicycle sharing system of Lisbon as an optimization problem to locate the stations and determine the fleet size through a Mixed Integer Linear Program (MILP). Correia and Antunes (2012) developed a mixed-integer programming model to maximize the profit of the one way car-sharing system. Using their model, the authors anticipated 75 possible depot locations for the Lisbon case study. Nair and Miller (2014) developed a bi-level mixed-integer program to compute the locations of the stations and their capacities considering a budget constraint with the goal of maximizing the revenue of the sharing system. Fricker and Gast (2016) studied users' random choice when there is no bike for renting or there is no available spot for returning bikes by defining a stochastic model. They computed the fleet size reducing the number of problematic stations. Yan et al. (2017) studied four NP-hard models under deterministic and stochastic demands. They defined locations of the stations and the fleet size and used a heuristic algorithm for solving the stochastic models. Moreover, these models can be considered as a Vehicle Routing Problem (VRP) and some papers that have studied VRP models can be introduced as Goli et al. (2018), Goli & Davoodi (2018), Tirkolaee et al. (2019), Sangaiah et al. (2019).

Some researchers studied the fleet size in more details considering the inventory of stations. Raviv and Kolka (2013) developed a model for calculating the initial inventory level of the bicycles with the aim to minimize the customer dissatisfaction. Users are dissatisfied when they face shortage of bicycles or shortage of vacant lockers. Jian et al. (2016) studied the New York Citi Bike system in determining the bike and dock allocations of the stations aiming to minimize the number of bike shortages for renting and dock shortages for returning bikes. They used heuristic methods based on simulation to solve the model.

In public bicycle sharing systems, because of various demand rates for different destinations, after a while the system may face imbalanced stations. In other words, users may face bike shortages for renting or dock shortages for returning the bicycles. There are concerned with rebalancing the stations by repositioning the bicycles. Static rebalancing bicycle problems (SRBP) were developed to rebalance stations during idle times. On the other hand, dynamic rebalancing problems (DRBP) were developed to reposition bicycles when the system is active. Erdogan et al. (2014) developed a model for the static case considering lower and upper bounds for inventory of the stations. They defined a single vehicle for delivering the bicycles between stations for rebalancing. A branch and cut algorithm was used to solve the integer programming model. Kadri et al. (2016) studied a static rebalancing problem in which each station was visited only once by the vehicle. Their model aimed to find the schedule of the vehicle so that the total waiting time in the disequilibrium situation was minimized. Bulhões et al. (2018) presented a model for the static case considering multiple vehicles with identical capacities and service time limits. Other works concerned with static rebalancing problems include Nair and Miller (2011), Raviv et al. (2013), Dell'Amico et al. (2014) and Valdes et al. (2016).

Pfrommer et al. (2014) defined a dynamic system, encouraging users to park rented bikes at nearby under-used stations to minimize the cost of repositioning bikes by the staff. They used the model for London's Barclays Cycle Hire scheme.

Regue and Recker (2014) presented an approach for repositioning with the goal of preventing unbalanced stations. They presented their model based on four core models including demand forecasting, station inventory, redistribution needs and a vehicle-routing model. Zhang et al. (2017) discussed their dynamic model by forecasting inventory levels and user arrivals for repositioning bicycles. They evaluated their mixed-integer model using a heuristic algorithm. Other researchers like Contardo et al. (2012), Chemla et al. (2013) and Ghosh and Varakantham (2017) also considered the dynamic case in their models.

There are various types of Public Bicycle Sharing Systems in the world and this paper discuses a model in which the customers predefine their destinations before renting the bikes using an application in the station. The goal of this manuscript is rejecting a part of requests to minimize the objective function which includes the total number of rejected demands. So, the proposed model tries to balance the inventory of the system by controlling the demands without using any extra vehicles to move the surplus bicycles.

Here, we are to use a proactive scenario to balance the stations. By preventing and rejecting demands for some destinations, the system may get the opportunity to reduce problematic stations such as stations having low inventories and fulfilled stations. The proposed policy will be enforced through defining response rates. Of course, having a more balanced inventory system helps to reduce the cost of repositioning. The objective function contains two parts in which the first one is the mean of the rejected demands due to the lack of available bikes for renting based on predefined response rates. The second part is defined as the mean users waiting for a vacant dock to return bikes. In the next section, the mathematical model is described and formulated. A genetic algorithm (GA) considering the mean value analysis (MVA) method will be proposed to solve the model in Section 3. Finally, numerical examples are worked through to illustrate the applicability of the proposed method.

2. Model description

Consider a bicycle sharing system in which users are allowed to rent bicycles from any available station and return them back to a destination station after a short travel. A customer arrives at station i following a Poisson process with the rate γ_i and requests a bike to destination j with probability r_{ij}. The customer's trip lasts according to an exponential distribution with rate $\beta_l$in which l is the node defined for the route from station i to station j. If the maximum response rate of the proposed destination is r'_{ij}, then the request is rejected with probability $(1 - r'_{ij})$. In this case, the user leaves station i without receiving a service. In other words, $r'_{ij} r_{ij} \gamma_i$ is considered as the new request rate for the proposed destination. Obviously, response rates are meaningful when station i is not empty and there is at least one bicycle for renting. Otherwise, customer leaves station i with probability 1. Stations have specified capacities and when a customer arrives at her destination, she should park the bicycle at a vacant dock and if there is no empty dock, she must wait until a dock becomes available. Response rates are used to control the inventory levels of different stations with the goal of balancing the system during the day as much as possible without using trucks for picking up and transferring bikes from overloaded stations to the ones with low inventories. The objective is to find the proper response rates to minimize the mean number of rejected renting requests and the mean number of customers waiting for vacant docks. The mathematical model will be described using the notation given in the following subsection.

2.1. Parameters

Parameters, indices and decision variables are defined as follows.

2.1.1. Indices and parameters

S: Set of station nodes (nodes of the Jackson Network with finite servers)

K: Number of stations

I: Set of route nodes (nodes of the Jackson Network with infinite servers) in which each route is shown as combination of two stations ($\binom{K}{2}$ nodes)

R: Set of nodes ($R = I \cup S$)

N: Total number of nodes ($N=K+\binom{K}{2}$)

i, j, l: Indices of set R with i=1,2,3,...,K showing the bicycle stations, $i=K+1, K+2, \ldots, K+(K-1)$ showing routes from station 1 to stations 2,3,...,K, respectively, continuing up to $i=K+(K-1)^2+1,\ K+(K-1)^2+2,\ \ldots,\ K+(K-1)^2+(K-1)$ showing routes from station K to stations 1,2,..,$K-1$, respectively

n_i: Number of available bicycles at node i ($i \in R$)

$\bar{n}$: Components vector for describing each state shown as $(n_1, n_2, \ldots, n_N)$

$\pi(n_i)$: Probability of presence of n_i bicycles at station i in the steady state (the state of the system after it has been in operation a long time)

$\pi_i(n_i, t)$: The probability of presence of n_i bicycles at station i in the steady state when the fleet size equals t

r_{ij}: Probability of requesting a bike from station i to station j ($i, j \in S$)

p_{il}: Elements of the route matrix P

p'_{il}: Elements of the second route matrix P'

F: Maximum capacity of the system, that is, $F = \sum_{\forall i \in S} s'_i$

γ_i: Mean of requests for renting bike in station i ($i \in S$)

β_i: Rate in an exponential distribution in which $\frac{1}{\beta_i}$ is the mean time for each bike to finish route i ($i \in I$)

c_i: Number of servers at node i; $c_i = 1, \forall i \in S$ and $c_i = F, \forall\ i \in I$

μ_i: Service rate at node i; $\mu_i = \begin{cases} F\beta_i, & \text{for } i \epsilon I \\ \gamma_i, & \text{for } i \in S \end{cases}$

λ_i: Total mean flow rate into node i ($i \in R$)

$\lambda_i(t)$: Total mean flow rate into node i when the fleet size equals t ($i \in R$)

L_i: Expected number of bikes at node i ($i \in R$)

$L_i(t)$: Expected number of bikes at node i when the fleet size equals t ($i \in R$)

W_i: The mean waiting time at node i ($i \in R$)

$W_i(t)$: Mean waiting time at node i when the fleet size equals t ($i \in R$)

s_i: Number of stocks at station i (the capacity of station i); ($i \in S$).

2.1.2. Decision variables

r'_{ij}: Response rate for renting a bike from station i to station j. $\left(1 - r'_{ij}\right)$ is the probability of rejecting a request in station i for going to station j ($i, j \in S$).

2.2. Network analysis

As there is a finite number of bicycles traveling between the stations, viewing the system from a bicycle perspective, a closed network system can be defined for the model. In this case, the network includes two groups of nodes. The first group contains rental stations and customer arrivals are considered as a virtual service. The second group consists of routes between the rental stations and the traveling time is interpreted as a virtual service. The number of servers at rental stations and routes are considered to be 1 and F, respectively. As the maximum number of bikes in the system equals F, the number of servers for the infinite server nodes is defined to be F. Figure 1 indicates the inputs and outputs of node 1 and their probabilities to show the relations between nodes of a network. Considering the response rates, the network relationships are accordingly depicted in Figure 2.

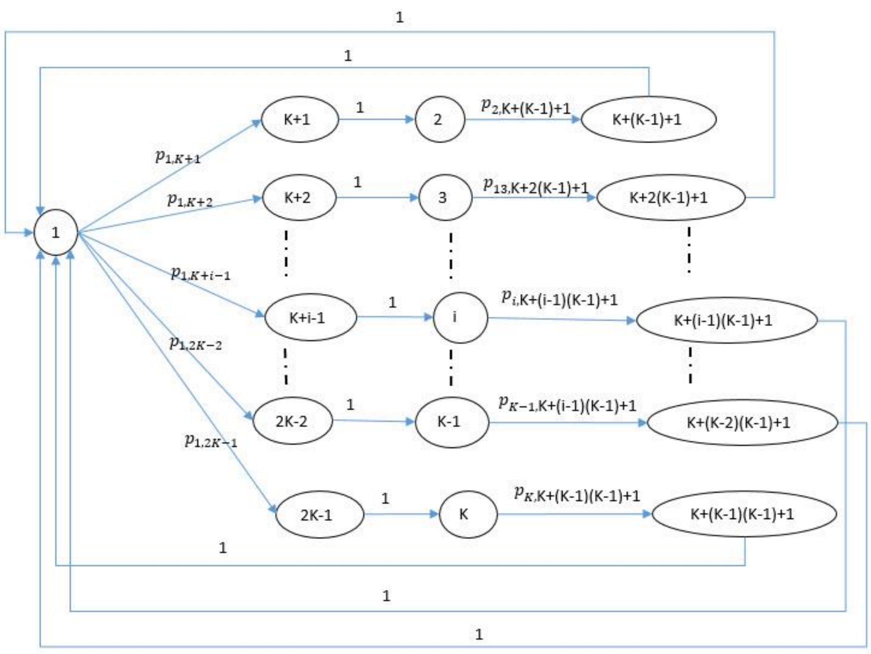


Figure 1. Inputs and outputs of node 1 without response rates to show the relations between nodes of the network

To obtain the matrix P, the following procedure is used note that "[ ]" is the symbol used for integral part and $i, j \in S$ and $l \in I$. This procedure calculates the probabilities of moving a bike between various nodes of the network including route nodes and station nodes. As an instance, the probability of moving a bicycle from one route node to another route node equals zero.

The probability of transferring a bicycle from station node i to the route node which is related to traveling from station i to station j equals r_{ij}. Finally, the probability of moving a bike from the proposed route station to station node j equals 1.

For i=1:K do
For l=K+1:K+K(K−1) do
For j=1:K do
If $l = K + (i-1)(K-1) + j - 1$ Or $l = K + (i-1)(K-1) + j$ Then
$p_{il} = r_{ij}$
End if
If $i = l - K - (K-1)\left[\frac{l-K-1}{K-1}\right]$ & $i < \left[\frac{l-K-1}{K-1}\right] + 1$ Then
$p_{li} = 1$
Else If $i = l - K - (K-1)\left[\frac{l-K-1}{K-1}\right]$ & $i \geq \left[\frac{l-K-1}{K-1}\right] + 1$ Then
$p_{l,i+1} = 1$
Else $p_{il} = 0$
End if
End do
End do
End do

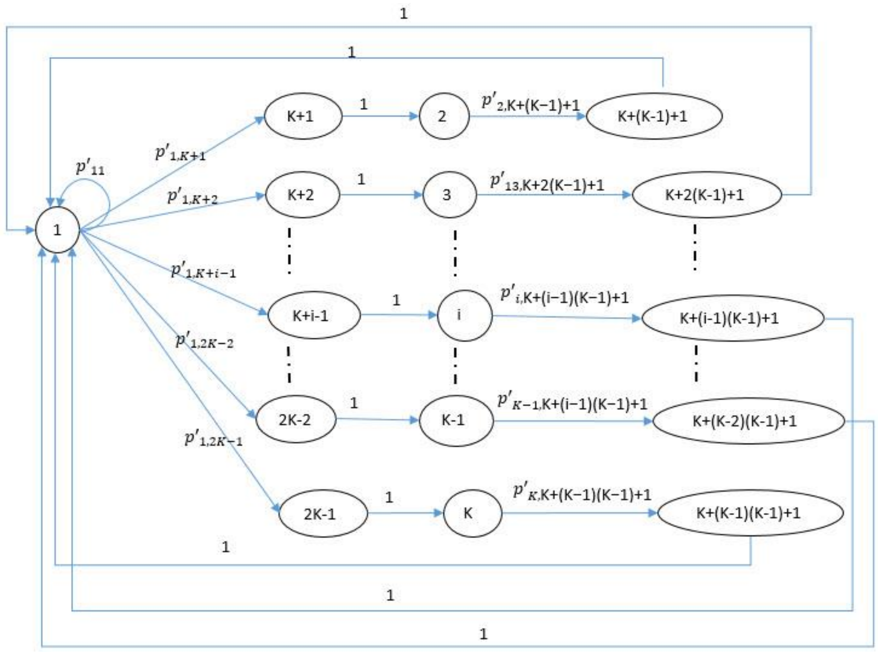


Figure 2. Inputs and outputs of node 1 considering response rates to show the relations between nodes of the network

Next, to obtain the matrix P', the following procedure is used:

For i=1:K do
For l=K+1:K+K(K−1) do
For j=1:K do
If $l = K + (i-1)(K-1) + j - 1$ Or $l = K + (i-1)(K-1) + j$ Then
$p'_{il} = r'_{ij}$
End if
If $i = l - K - (K-1)\left[\frac{l-K-1}{K-1}\right]$ & $i < \left[\frac{l-K-1}{K-1}\right] + 1$ Then
$p'_{li} = 1$
Else If $i = l - K - (K-1)\left[\frac{l-K-1}{K-1}\right]$ $i \geq \left[\frac{l-K-1}{K-1}\right] + 1$ Then
$p'_{l,i+1} = 1$
Else If i=l & $i \in S$ Then
$p'_{il} = 1 - \sum_{\forall j, j \neq i} p'_{ij}$
Else $p'_{il} = 0$
End if
End do
End do
End do
.

Table 1 defines different possible states and Figure 3 shows the rate transition diagram to depict the relations between the states.

Table 1. State descriptions

State	Simplified Notation
$(\boldsymbol{n_1}, \boldsymbol{n_2}, \dots, \boldsymbol{n_i}, \boldsymbol{n_j}, \dots, \boldsymbol{n_{w+K}})$	$\bar{n}$
$(\boldsymbol{n_1}, \boldsymbol{n_2}, \dots, \boldsymbol{n_i} + \boldsymbol{1}, \boldsymbol{n_j} - \boldsymbol{1}, \dots, \boldsymbol{n_{w+K}})$	$\bar{n}; i^+, j^-$

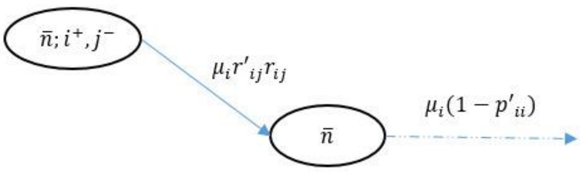


Figure 3. Rate transition diagram which shows the relations between the states

The rate of requests in station *i* for going to station *j* is$r'_{ij} r_{ij} \gamma_i$ $(i, j \in S)$. As long as the inventory of station *i* equals zero, all of its requests are rejected. Otherwise, requests in station *i* for destination *j* are responded with probabilityr'_{ij}. In other words, when there is at least one bike for renting at the original station *i*, then requests from station *i* to station *j* are rejected with probability$(1 - r'_{ij})$. This policy is defined to manage the inventory of the system with the goal of minimizing dissatisfaction by reducing the number of stations having lack of inventory and fulfilled stations. Although some renting requests are rejected, lack of empty docks for returning bicycles gets reduced and the overall response to the requests is expected to increase. Response rates lead to sending bikes to stations with high levels of requests by rejecting and reducing requests of other destinations.

2.3. Objective function

According to our indications in the previous section, the objective function can be written as follows:

Min $\quad \sum_{i\in S}\sum_{n_i=s_i}^{F}(n_i - s_i)\,\pi(n_i) + \sum_{i\in S}\sum_{j\in I}\gamma_i {p'}_{ij}\,\pi(n_i = 0) + \sum_{i\in S}\gamma_i {p'}_{ii}\ \pi(n_i \geq 1),$ (1)

Where it is to minimize the mean amount of inventory being more than the capacity of stations plus the mean number of requests that are denied in the steady state. The first term gives the mean number of bicycles having to wait for vacant docks, and the second term gives the mean number of requests not being satisfied because the inventory level of the station equals zero and the third term defines the mean number of requests not being satisfied when requests are rejected according to the response rates.

3. Solution method

PBSS models which are developed for determining the fleet size of the system based on a queueing network, are known as high complicacy models due to being categorized as an NP-hard problem. So, in this paper a meta-heuristic algorithm is developed to solve the proposed model. The Mean Value Analysis (MVA) is an appropriate approximation method for solving closed queuing networks. A description of this method can be found in Bruell and Balbo (1980). Here, we develop a genetic algorithm in which MVA is used for fitness calculations. The chromosome is defined as follows to show the response rates for different destinations:

$$\boldsymbol{Chromosome}: \quad \begin{bmatrix} \boldsymbol{r'_{12}} & \boldsymbol{r'_{13}} & \cdots & \boldsymbol{r'_{1K}} \\ \boldsymbol{r'_{21}} & \boldsymbol{r'_{23}} & \cdots & \boldsymbol{r'_{2K}} \\ \cdots & \cdots & & \cdots \\ \boldsymbol{r'_{K1}} & \boldsymbol{r'_{K2}} & \cdots & \boldsymbol{r'_{K,K-1}} \end{bmatrix},$$

Where each row contains the response rates of the corresponding route from an original station to all other possible destinations. More specifically, the first row shows the response rates for destinations starting from the first station (node 1) to $K-1$ other possible stations (nodes:$K+1, K+2, \dots, K+K-1$) and the last row shows the response rates for destinations which start from the last station (node K) to $K-1$ possible destination (nodes: $K+(K-1)^2+1, \dots, K+(K-1)^2+2, \dots, K+(K-1)^2+K-1$).

The steps of our proposed genetic algorithm can now be described as follows:

Produce the initial population randomly according to the chromosome structure specified above.

Set iteration number to 1 (*It*=1).

Calculate P and P' for each chromosome according to the procedures given in Section 2.2.

{Calculate the amount of fitness for each chromosome using the MVA method}

Solve weighted traffic equation $\upsilon_j = \sum_{i=1}^{N}\upsilon_i {p'}_{ij}\ (i,j \in R)$, where $\upsilon_i = \frac{\lambda_i}{\lambda_l}$ (υ_l is normalized and is equal to 1 and l is selected from the set R)

For i=1:N do

$$L_i(0) = 0$$
$$\pi_i(0,0) = 1$$
$$\pi_i(n_i, 0) = 0 \ ; \ n_i \neq 0$$

End do

For t=1:F do

For i=1:K do

$$W_i(t) = \frac{1}{c_i \mu_i}\left(1 + L_i(t-1) + \sum_{n_i=0}^{c_i-2} (c_i - 1 - n_i)\pi_i(n_i, t-1)\right)$$

$; c_i = F(\forall i \in I) \ and \ c_i = 1(\forall i \in S)$

$$\lambda_l(t) = \frac{t}{\sum_{i=1}^{N} \vartheta_i W_i(t)}$$
$$\lambda_i(t) = \lambda_l(t)\vartheta_i \ ; \ i \neq l$$
$$L_i(t) = \lambda_i(t) W_i(t)$$

For $n_i = 1 : t$ do

$$\pi_i(n_i, t) = \frac{\lambda_i(t)}{\mu_i a_i(n_i)} \pi_i(n_i - 1, t-1) \ ; \ a_i(n_i) = \begin{cases} n_i, & n_i \leq c_i \\ c_i, & n_i \geq c_i \end{cases}$$

End do

End do

End do

{Develop a new population}

Crossover operation: a random crossover mask is used to determine which components of the parents can be selected for generating the new chromosomes. The number of the new generated chromosomes is defined according to the crossover percentage. Figure 4 shows an example of the crossover operation.

$$\boldsymbol{Parents}: \quad \begin{bmatrix} \boldsymbol{r'}_{12} & \boldsymbol{r'}_{13} & \boldsymbol{r'}_{14} \\ \boldsymbol{r'}_{21} & \boldsymbol{r'}_{23} & \boldsymbol{r'}_{24} \\ \boldsymbol{r'}_{31} & \boldsymbol{r'}_{32} & \boldsymbol{r'}_{34} \\ \boldsymbol{r'}_{41} & \boldsymbol{r'}_{42} & \boldsymbol{r'}_{43} \end{bmatrix} \quad , \quad \begin{bmatrix} \boldsymbol{r''}_{12} & \boldsymbol{r''}_{13} & \boldsymbol{r''}_{14} \\ \boldsymbol{r''}_{21} & \boldsymbol{r''}_{23} & \boldsymbol{r''}_{24} \\ \boldsymbol{r''}_{31} & \boldsymbol{r''}_{32} & \boldsymbol{r''}_{34} \\ \boldsymbol{r''}_{41} & \boldsymbol{r''}_{42} & \boldsymbol{r''}_{43} \end{bmatrix}$$

$$\boldsymbol{Random\ crossover\ mask}: \begin{bmatrix} \mathbf{1} & \mathbf{0} & \mathbf{1} \\ \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{1} & \mathbf{0} \\ \mathbf{1} & \mathbf{1} & \mathbf{1} \end{bmatrix}$$

$$\boldsymbol{Offspring}: \quad \begin{bmatrix} \boldsymbol{r'}_{12} & \boldsymbol{r''}_{13} & \boldsymbol{r'}_{14} \\ \boldsymbol{r'}_{21} & \boldsymbol{r''}_{23} & \boldsymbol{r''}_{24} \\ \boldsymbol{r''}_{31} & \boldsymbol{r'}_{32} & \boldsymbol{r''}_{34} \\ \boldsymbol{r'}_{41} & \boldsymbol{r'}_{42} & \boldsymbol{r'}_{43} \end{bmatrix} \quad , \quad \begin{bmatrix} \boldsymbol{r''}_{12} & \boldsymbol{r'}_{13} & \boldsymbol{r''}_{14} \\ \boldsymbol{r''}_{21} & \boldsymbol{r'}_{23} & \boldsymbol{r'}_{24} \\ \boldsymbol{r'}_{31} & \boldsymbol{r''}_{32} & \boldsymbol{r'}_{34} \\ \boldsymbol{r''}_{41} & \boldsymbol{r''}_{42} & \boldsymbol{r''}_{43} \end{bmatrix}$$

Figure 4. An example for crossover operation based on step 5.1 (*K*=4, number of nodes in network is equal to 16).

Mutation operation: a random vector is generated and the elements of the selected chromosome whose corresponding elements in the random vector are less than the mutation percentage are regenerated. Figure 5 shows an example of the mutation operation.

Elitism operation: the best chromosomes of current population is sent to the next population considering the elitism percentage.

If the *It* is less than a pre-specified maximum number (*MaxIt*), let *It*=*It*+1 and go to step 3, else go to step 7.

Select the best chromosome of the final population as the final solution.

$$\textbf{Selected chromosome}: \begin{bmatrix} r'_{12} & r'_{13} & r'_{14} \\ r'_{21} & r'_{23} & r'_{24} \\ r'_{31} & r'_{32} & r'_{34} \\ r'_{41} & r'_{42} & r'_{43} \end{bmatrix} \quad \textbf{Random matrix}: \begin{bmatrix} 0.004 & 0.1 & 0.35 \\ 0.003 & 0.86 & 0.023 \\ 0.76 & 0.056 & 0.012 \\ 0.45 & 0.034 & 0.08 \end{bmatrix}$$

$$\textbf{Offspring}: \begin{bmatrix} r''_{12} & r'_{13} & r'_{14} \\ r''_{21} & r'_{23} & r''_{24} \\ r'_{31} & r'_{32} & r''_{34} \\ r'_{41} & r''_{42} & r'_{43} \end{bmatrix}$$

Figure 5. An example for mutation operation considering 0.05 for mutation percentage based on step 5.2 (*K*=4, and the number of nodes in the network is equal to 16).

All the parameters of the proposed genetic algorithm are tuned using the Taguchi method (see Byme and Taguchi, 1987) and the tuned amounts for crossover percentage, mutation percentage and population size are obtained to be 0.85, 0.005 and 131, respectively. Figure 6 shows the proposed genetic algorithm.

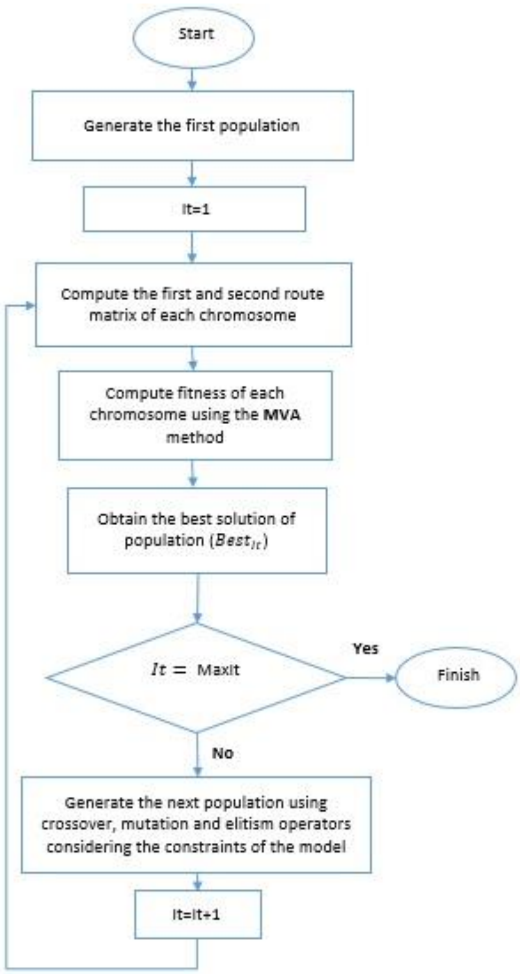


Figure 6. The proposed Genetic Algorithm

4. Numerical examples

Consider a small bicycle sharing system with 3 stations having arrival rates 47, 24 and 32 units per hour, respectively. To explain it in the form of a Jackson network, 6 more nodes are required to be defined as route nodes. User who arrives at station 1 rents a bicycle to go to station 2 with probability 0.3 and to station 3 with probability 0.7. Other probabilities of renting bikes from station 2 to stations 1 and 3, station 3 to stations 1 and 2 are 0.4, 0.6, 0.2 and 0.8, respectively. The route matrix is as follows:

$$P=\begin{bmatrix} 0 & 0 & 0 & 0.3 & 0.7 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.4 & 0.6 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.2 & 0.8 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Each station contains 18 docks. The time required for traveling each route follows on exponential distribution with rates 10, 12, 8, 14, 5 and 8 for nodes 4 through 9, respectively. As the fleet size is equal to 54, service rates can be defined as the following vector for nodes 1 to 9:

$\mu = [47 \;\; 24 \;\; 32 \;\; 540 \;\; 648 \;\; 432 \;\; 756 \; 270 \;\; 432].$

The problem was solved by the MVA method and the objective function value was obtained 66.5307, the mean number of rejected requests was 39.9303 and the mean number of users waiting for vacant docks was 26.6003. After tuning the parameters of the proposed GA by the Taguchi method, crossover percentage, mutation percentage and population size were obtained to be 0.85, 0.005 and 131, respectively. Using the proposed GA, the objective function value is 48.3033 which means that the sum of the mean number of rejected renting requests and the mean number of users waiting for parking bikes is reduced about 18.2274 units at the steady state. The mean number of users waiting and rejected renting demands were 13.6876 and 34.6156, respectively. The following chromosome shows the final solution:

$$GA\ Final\ Solution: \quad \begin{bmatrix} 0.012 & 0.521 \\ 1 & 1 \\ 1 & 0.934 \end{bmatrix}.$$

Therefore, the matrix P' is:

$$P'=\begin{bmatrix} 0.6317 & 0 & 0 & 0.0036 & 0.3647 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.4 & 0.6 & 0 & 0 \\ 0 & 0 & 0.0528 & 0 & 0 & 0 & 0 & 0.2 & 0.7472 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Figure 7 shows how the response rates have changed the elements of the route matrix for each station. As an instance, by rejecting 98.8 percent of the requests from station 1 to station 2, 47.9 percent of the requests from station 1 to station 3 and 6.6 percent of the demands from station 3 to station 2, the system's total mean number of rejected requests and its total mean number of lack of vacant docks for returning bikes are improved in the steady state. Table 2 shows the objective function values of the problem before and after implementing the response rates. Although the system is rejecting some demands according to predefined response rates, it could reduce the total mean number of unsatisfied users in the steady state.

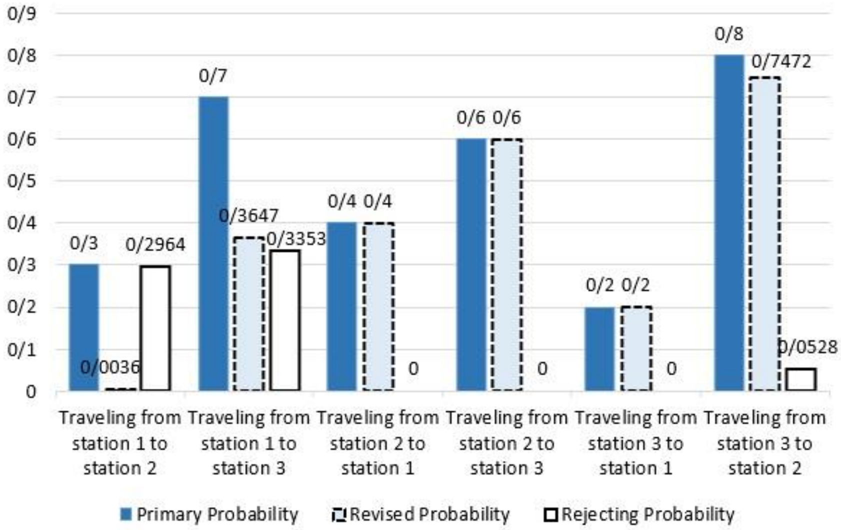


Figure 7. Change of the probabilities for different routes after exerting new policy for routes *ij*

(Station *i* to station *j*)

For more description, as it is depicted in figure 7, for traveling from station 1 to station 2, the primary probability of sending a bike from station 1 to station 2 is 0.3 and 98.8 percent of the requests for this route will be rejected. So, the rejecting probability which here means the probability of sending a bike from station 1 to itself (rejecting the demand) can be calculated as 0.3×0.988 which equals 0.2964 and the revised probability is 0.0036 which is the new probability of sending a bike from station 1 to station 2.

Table 2. Comparing the results obtained by GA with the ones obtained by the general condition (MVA) for the 3 stations example in the steady state

	MVA (General Condition)	GA (Considering response rates)	Improvement compared to the general condition
Mean number of rejected requests	39.9303	34.6156	5.3147
Mean number of users waiting	26.6003	13.6876	12.9127
Objective function value	66.5307	48.3033	18.2274
Mean number of satisfied renting requests	63.0697	68.3844	5.3147

It should be noted that by increasing the number of stations, complexity of the system increases and the proposed GA is more efficient in reducing the number of dissatisfied users. Table 3 gives a summary of different numerical examples considering various fleet sizes. We can clearly observe the effectiveness of GA in all the examples.

To define a systematic rule for rejecting demands based on response rates, users can be categorized in different classes according to various indices such as Recency, Frequency and Neighborhood. Each customer has a membership code and can be recognized via a cell phone or a smart card. After grouping users in different classes, the percentage of applicants for each station can be obtained considering different classes of users. For instance, consider that users of station 1 are categorized into five groups with group 1 through 5 respectively having 5, 10, 25, 32 and 28 percentage of the total users. Entrance of different classes is random and the response rate for renting a bike in station 1 for going to station 3 is 52.1 percent. Then, the requests of this route in station 1 are only satisfied for classes 1 to 3 and 12.1 percent of the fourth class. Other renting requests in station 1 for going to station 3 are rejected.

Table 3. Random examples

Example	Number of stations	Fitness of MVA (General Condition)	Fitness of GA (Considering response rates)	Improvement compared to the general condition
1	3	130	119	11
2	6	302	268	34
3	8	511	376	135
4	10	656	487	169
5	12	886	613	273
6	18	1189	886	303
7	22	1768	1567	201
8	25	1154	885	269
9	28	1322	976	346
10	30	1562	1223	339

5. Conclusion

We introduced response rates with the goal of balancing the inventory of a public bicycle sharing system. The policy was to reject some parts of different renting requests to avoid sending bicycles to destinations lacking vacant docks and saving bikes for destinations having low inventories. Although some parts of renting requests were rejected by the system, the inventories of stations were balanced so that the total sum of the mean number of rejected requests and the mean number of users waiting for empty docks were reduced in the steady state. Using the Mean Value Analysis method, a genetic algorithm was developed for solving the problem and obtaining proper response rates. After tuning parameters of the proposed algorithm by the Taguchi method, different numerical examples were solved. The results showed the proposed policy to be effective for improving the operation of the public bicycle sharing system and reducing the users' dissatisfaction. The model can be extended considering the fleet size and the capacities of the stations as decision variables.

Acknowledgements

The first two authors thank Mazandaran University of Science and Technology. The third author thanks Sharif University of Technology and the forth author thanks Amirkabir University of Technology for supporting this work.

References

Alvarez-Valdes, R., Belenguer, J. M., Benavent, E., Bermudez, J. D., Muñoz, F., Vercher, E., and Verdejo, F., (2016). "Optimizing the level of service quality of a bike-sharing system", *Omega*, Vol. 62, pp. 163-175.

Bulhões, T., Subramanian, A., Erdoğan, G., and Laporte, G., (2018). "The static bike relocation problem with multiple vehicles and visits", *European Journal of Operational Research*, Vol. 264, No. 2, pp. 508-523.

Bruell, S. C., and Balbo, G., (1980). "Computational Algorithms for Closed Queueing Networks". Elsevier North-Holland, pp. 1-206.

Bryne, D. M., and Taguchi, S., (1987). "Taguchi approach to parameter design". Quality progress, 19-26.

Chemla, D., Meunier, F., Pradeau, T., Calvo, R. W., & Yahiaoui, H., (2013). "Self-service bike sharing systems: simulation, repositioning, pricing".

Contardo, C., Morency, C., and Rousseau, L. M., (2012). Balancing a dynamic public bike-sharing system (Vol. 4). Montreal: Cirrelt.

De Almeida Correia, G. H., and Antunes, A. P., (2012). "Optimization approach to depot location and trip selection in one-way carsharing systems", *Transportation Research, Part E: Logistics and Transportation Review*, Vol. 48, No. 1, pp. 233-247.

Dell'Amico, M., Hadjicostantinou, E., Iori, M., and Novellani, S., (2014). "The bike sharing rebalancing problem: Mathematical formulations and benchmark instances", *Omega*, Vol. 45, pp. 7-19.

Erdoğan, G., Laporte, G., and Calvo, R. W., (2014). "The static bicycle relocation problem with demand intervals", *European Journal of Operational Research*, Vol. 238, No. 2, pp. 451-457.

Fricker, C., and Gast, N., (2016). "Incentives and redistribution in homogeneous bike-sharing systems with stations of finite capacity", *Euro Journal on Transportation and Logistics*, Vol. 5, No. 3, pp. 261-291.

George, D. K., and Xia, C. H., (2011). "Fleet-sizing and service availability for a vehicle rental system via closed queueing networks", *European Journal of Operational Research*, Vol. 211, No. 1, pp. 198-207.

Ghosh, S., and Varakantham, P., (2017). Incentivizing the use of bike trailers for dynamic repositioning in bike sharing systems.

Goli, A., Aazami, A., and Jabbarzadeh, A., (2018). "Accelerated cuckoo optimization algorithm for capacitated vehicle routing problem in competitive conditions", *Int J Artif Intell*, Vol. 16, No. 1, pp. 88-112.

Goli, A., and Davoodi, S. M. R., (2018). "Coordination policy for production and delivery scheduling in the closed loop supply chain", *Production Engineering*, Vol. 12, No. 5, pp. 621-631.

Jian, N., Freund, D., Wiberg, H. M., and Henderson, S. G., (2016, December). "Simulation optimization for a large-scale bike-sharing system". In Proceedings of the 2016 Winter Simulation Conference. IEEE Press, pp. 603-613.

Kadri, A. A., Kacem, I., and Labadi, K., (2016). "A branch-and-bound algorithm for solving the static rebalancing problem in bicycle-sharing systems", *Computers & Industrial Engineering*, Vol. 95, pp. 41-52.

Lin, J. R., and Yang, T. H., (2011). "Strategic design of public bicycle sharing systems with service level constraints", *Transportation Research, Part E: Logistics and Transportation Review*, Vol. 47, No. 2, pp. 284-294.

Martinez, L. M., Caetano, L., Eiró, T., and Cruz, F., (2012). "An optimisation algorithm to establish the location of stations of a mixed fleet biking system: an application to the city of Lisbon", *Procedia-Social and Behavioral Sciences*, Vol. 54, pp. 513-524.

Nair, R., and Miller-Hooks, E., (2011). "Fleet management for vehicle sharing operations", *Transportation Science*, Vol. 45, No. 4, pp. 524-540.

Nair, R., and Miller-Hooks, E., (2014). "Equilibrium network design of shared-vehicle systems", *European Journal of Operational Research*, Vol. 235, No. 1, pp. 47-61.

Pfrommer, J., Warrington, J., Schildbach, G., and Morari, M., (2014). "Dynamic vehicle redistribution and online price incentives in shared mobility systems", *IEEE Transactions on Intelligent Transportation Systems*, Vol. 15, No. 4, pp. 1567-1578.

Raviv, T., & Kolka, O., (2013). "Optimal inventory management of a bike-sharing station", *IIE Transactions*, Vol. 45, No. 10, pp. 1077-1093.

Raviv, T., Tzur, M., and Forma, I. A., (2013). "Static repositioning in a bike-sharing system: models and solution approaches", *EURO Journal on Transportation and Logistics*, Vol. 2, No. 3, pp. 187-229.

Regue, R., and Recker, W., (2014). "Proactive vehicle routing with inferred demand to solve the bikesharing rebalancing problem", *Transportation Research, Part E: Logistics and Transportation Review*, Vol. 72, pp. 192-209.

Sangaiah, A. K., Tirkolaee, E. B., Goli, A., and Dehnavi-Arani, S., (2019). "Robust optimization and mixed-integer linear programming model for LNG supply chain planning problem", *Soft Computing*, pp. 1-21.

Tirkolaee, E. B., Goli, A., and Weber, G. W., (2019). "Multi-objective Aggregate Production Planning Model Considering Overtime and Outsourcing Options Under Fuzzy Seasonal Demand", In Advances in Manufacturing II (pp. 81-96). Springer, Cham.

Yan, S., Lin, J. R., Chen, Y. C., and Xie, F. R., (2017). "Rental bike location and allocation under stochastic demands", *Computers & Industrial Engineering*, Vol. 107, pp. 1-11.

Yang, X. H., Cheng, Z., Chen, G., Wang, L., Ruan, Z. Y., and Zheng, Y. J., (2018). "The impact of a public bicycle-sharing system on urban public transport networks", *Transportation Research, Part A: Policy and Practice*, 107, 246-256.

Zhang, D., Yu, C., Desai, J., Lau, H. Y. K., and Srivathsan, S., (2017). "A time-space network flow approach to dynamic repositioning in bicycle sharing systems", *Transportation Research, Part B: Methodological*, Vol. 103, pp. 188-207.

This article can be cited: Maleki Vishkaei, B., Mahdavi, I., Mahdavi-Amiri N., Khorram E., (2020). " Balancing public bicycle sharing system by defining response rates for destinations ", *Journal of Industrial Engineering and Management Studies*, Vol. 7, No. 1, pp. 19-34.