27 September 2020

EXACT LOADING AND UNLOADING STRATEGIES FOR THE STATIC MULTI-VEHICLE BIKE REPOSITIONING PROBLEM

W.Y. Szeto*, C.S. Shui
Department of Civil Engineering, The University of Hong Kong
The University of Hong Kong Shenzhen Institute of Research and Innovation
*Email: ceszeto@hku.hk

ABSTRACT

This study investigates a bike repositioning problem (BRP) that determines the routes of the repositioning vehicles and the loading and unloading quantities at each bike station to firstly minimize the positive deviation from the tolerance of total demand dissatisfaction (TDD) and then service time. The total demand dissatisfaction of a bike-sharing system in this study is defined as the sum of the difference between the bike deficiency and unloading quantity of each station in the system. Two service times are considered: the total service time and the maximum route duration of the fleet. To reduce the computation time to solve the loading and unloading sub-problem of the BRP, this study examines a novel set of loading and unloading strategies and further proves them to be optimal for a given route. This set of strategies is then embedded into an enhanced artificial bee colony algorithm to solve the BRP. The numerical results demonstrate that a larger fleet size may not lead to a lower total service time but can effectively lead to a lower maximum route duration at optimality. The results also illustrate the trade-offs between the two service times, between total demand dissatisfaction and total service time, and between the number of operating vehicles provided and the TDD. Moreover, the results demonstrate that the optimal values of the two service times can increase with the TDD and introducing an upper bound on one service time can reduce the optimal value of the other service time.

Keywords: static bike repositioning; exact loading and unloading strategies; tolerance of total demand dissatisfaction

1 INTRODUCTION

Cycling has been widely recognized as an economical and environmentally friendly mode of transport. Particularly in overseas countries, citizens are very much encouraged to travel by cycling in short-distance so that the number of motorized trips can be reduced, with a view to reducing pollution from motorized vehicles. To make cycling more convenient and efficient, the concept of bike sharing was introduced to allow the rental use of bicycles at specified stations within a city and return at any other stations whenever a short-distance trip is required (Raviv et al., 2013). Whereas bike-sharing systems are commonly run by the public sector, they can also be run successfully overseas by commercial parties to gain revenue or by a public-private partnership. Bike sharing is currently very popular worldwide. As of 2 November 2017, public bike-sharing systems were available in about 1,488 cities and included approximately 18,740,100 bicycles around the world (Meddin and DeMaio, 2017).

The main challenge in operating a bike-sharing system is that the numbers of bikes required at some stations are often insufficient to satisfy the corresponding cycling demand. With a fixed number of bikes in the bike-sharing system, the operator always needs to transfer the bikes from bike surplus stations to bike deficient stations by trucks to reduce unsatisfied cycling demand. This problem is called a public bike repositioning problem (BRP), which determines optimal truck routes and the loading and unloading activities of the trucks, subject to various constraints including vehicle, station, and operational constraints. Because of its unique characteristics and practical importance, the BRP has attracted the interest of many researchers in recent years. The BRP can be modeled as either a static or dynamic optimization problem (Raviv et al., 2013). The static problem considers night-time scenarios where the demand is low or the system is closed, while the dynamic problem considers daytime scenarios that take real-time usage of the system into account. Table 1 has summarized the BRP publications according to the operational scenario, the number of operating trucks used, problem objectives, and whether loading and unloading time is considered in the travel time/cost in the objective function. In terms of operational scenarios, the vast majority of the BRP studies address the static problem but only a few studies have tackled the dynamic problem (Caggiani and Ottomanelli, 2012; Contardo et al., 2012; Kloimüllner, et al., 2014; Regue and Recker, 2014; Brinkmann et al., 2015; Zhang et al., 2016; Shui and Szeto, 2017). In terms of the number of trucks used, both single and multiple vehicle problems are studied. In terms of formulations, multiple vehicle repositioning problems are straightforward extensions of single vehicle repositioning problems. However, it is more realistic to consider multiple vehicle repositioning problems for large-scale bike-sharing systems.

Table 1 Summary of the characteristics of the BRP in the existing literature

References	**Scenario**	**# of vehicles**	**Objectives**	**Loading & unloading time consideration in the travel time/cost in the objective function**
Benchimol et al. (2011)	Static	1	Minimize total travel cost	No
Caggiani and Ottomanelli (2012)	Dynamic	≥1	Minimize total relocation and lost user cost	Not applicable (NA)
Contardo et al. (2012)	Dynamic	≥1	Minimize total unmet demand	NA
Lin and Chou (2012)	Static	≥1	Minimize total transportation time or distance	No
Chemla et al. (2013)	Static	1	Minimize total travel cost	No
Di Gaspero et al. (2013a, 2016), Raidl et al. (2013), Rainer-Harbach et al. (2013, 2015)	Static	≥1	Minimize the weighted sum of the total absolute deviation from the target number of bikes, the total number of loading/unloading quantities, and the overall time required for all routes	Yes with an approximation*
Nair et al. (2013)	Static	1	Minimize total redistribution cost	Yes
Raviv et al. (2013), Forma et al. (2015)	Static	≥1	Minimize the weighted sum of total travel time and penalty cost	No
Angeloudis et al. (2014)	Static	≥1	Minimize total travel time	Yes, but the time is fixed duration per station visit
Dell' Amico et al. (2014, 2016)	Static	≥1	Minimize total travel time	No
Erdoğan et al. (2014)	Static	≥1	Minimize total travel and handling cost	Yes
Ho and Szeto (2014)	Static	1	Minimize total penalty cost	NA
Kloimüllner et al. (2014)	Dynamic	≥1	Minimize the weighted sum of unfulfilled demands, the absolute deviation from the target fill level, the total number of loading and unloading quantities, and total driving time	Yes with an approximation*

Regue and Recker (2014)	Dynamic	1	Minimize the weighted sum of total normalized travel time involved in going to buffer stations and the utilities gained by visiting stations with large inefficiency currently or stations with a neighborhood of stations expected to have inefficiencies in future time steps	No
Brinkmann et al. (2015)	Dynamic	≥1	Minimize the number of due date violations	NA
Alvarez-Valdes et al. (2016)	Static	≥1	Minimize the weighted sum of the total service time and the coefficient of variations of the duration of all routes	Yes but the loading and unloading times of a bike are the same
Kadri et al. (2016)	Static	1	Minimize the sum of the product of the deviation of the station inventory level from the acceptable level and the station imbalance time	No
Li et al. (2016)	Static	1	Minimize the sum of travel, imbalance, substitution, and occupancy costs	No
Szeto et al. (2016)	Static	1	Minimize the weighted sum of the total number of unsatisfied customers and the vehicle's total operational time	Yes with an approximation*
Zhang et al. (2016)	Dynamic	≥1	Minimize total vehicle travel cost plus the expected user dissatisfaction in the system	No
Cruz et al. (2017)	Static	1	Minimize route cost	No
Ho and Szeto (2017)	Static	≥1	Minimize the weighted sum of total travel time and penalty cost	No
Schuijbroek et al. (2017)	Static	≥1	Minimize maximum tour length	NA
Shui and Szeto (2017)	Dynamic	≥1	Minimize the weighted sum of total unmet demand and the fuel and CO_2 emission cost of the operating vehicle of each design interval	NA
This study	Static	≥1	Minimize first the positive deviation from the tolerance of total demand dissatisfaction and then service time in form of the total service time or maximum service time of all vehicles	Yes

* The total loading and unloading time at each station is approximated by the time required for an average number of loading and unloading operations or by other estimation.

In terms of problem objectives, common components are (1) travel time or related measures such as travel distance or cost, and (2) total demand dissatisfaction (e.g., total number of users who fail to get a bike at stations) in a bike-sharing system, in the form of unmet demand, penalty cost, or the deviation from the target inventory level. Whereas the first one is the key concern of private operators, the second one is a societal benefit measure and should be included in the objective by the government as the operator. Ideally, the total demand dissatisfaction of a bike-sharing system should be zero. In practice, some tolerance may be allowed. Although total demand dissatisfaction is crucial in any public bike system from the perspective of the society, solely focusing on the total demand dissatisfaction of the system does not guarantee that the final service routes can have the lowest service time (including in-vehicle travel time and loading and unloading times). Loading and unloading times also contribute to the operation hours of truck drivers, which have financial implications for the operator. It is important to capture the loading and unloading times accurately to minimize the operation cost for the operator (even if the operator is the public sector) because the cost can be viewed as a negative benefit (or a negative profit), and hence minimizing the operation cost can be viewed as maximizing the benefit. However, most of the preceding studies that capture travel time in the objective function do not use *actual* loading and unloading times in the travel time calculation. Moreover, those studies that use the actual loading and unloading times in travel time calculation do not consider the total demand dissatisfaction of the system simultaneously. Furthermore, multiple optimal solutions with the same minimum total demand dissatisfaction may exist. Using service time minimization as a second priority level objective helps the operator choose a solution among those optimal solutions with respect to the first priority level objective.

To address these issues, this paper proposes a new repositioning problem that minimizes firstly the positive deviation from the tolerance of the total demand dissatisfaction (TDD) of the system and then the service time of the vehicles (including actual loading and unloading times). The total demand dissatisfaction of a bike-sharing system in this study is defined as the sum of the difference between the bike deficiency and unloading quantity of each station in the system. The pre-determined parameter, TDD, is introduced into the first priority level objective function to allow a certain degree of bike deficiency in the system. Using this objective function allows us to derive and analyze simple loading and unloading strategies to determine *exact* pickup and drop-off quantities at each station. The service time of the vehicles required by the second priority level objective function is depicted by one of the two measures: the total service time and maximum route duration of all vehicles. The two measures result in two main models with their only difference in the measure used in the second priority level objective function.

It is noted that having too many excess bikes at some stations may result in inadequate parking spaces. To consider the space inventory required at each station for bike returns, unlike Nair and Miller-Hooks (2011) and Raviv et al. (2013) that implicitly considered and explicitly modeled the space inventory required in the penalty function and the level of service constraint respectively, this paper introduces a new approach to capture the target space inventory level (i.e., the number of slots reserved for parking) in the computation of the bike surplus and deficiency at each station.

To solve the proposed problem, the enhanced artificial bee colony (EABC) algorithm is used to determine vehicle routes and exact loading and unloading strategies are incorporated into the algorithm to determine pickup and delivery quantities at each station in each given route obtained from the algorithm. Based on the proposed solution method, test instances in the literature are used to illustrate the performance of the method and the properties of the problem.

To summarize, the main contributions of this study are the following:

1. We introduce the concept of the TDD and a new BRP that determines the vehicle routes and the loading and unloading quantities at each bike station to minimize first the positive deviation from TDD and then the total service time of all the vehicles; we also introduce the problem variant based on maximum route duration rather than total service time.
2. We propose and examine simple loading and unloading strategies and further prove them to be optimal solutions to the loading and unloading sub-problem. Using these strategies in the solution

process avoids the need of solving the time-consuming loading and unloading sub-problem as a linear program or maximum flow problem in each iteration (e.g., Rainer-Harbach et al., 2015). The resultant BRP can then be converted into a routing problem that is very similar to classical many-to-many pickup and delivery problems so that existing routing algorithms for the pickup and delivery problems can be used to solve our proposed problem.

3. We examine the properties of our proposed problem. In particular, we give propositions on loading and unloading strategies, investigate the trade-off between the TDD and the two service time measures, and illustrate the effect of the fleet size on the total demand dissatisfaction of the system and the two measures.

The remainder of the paper is organized as follows. Section 2 depicts and formulates the studied problem. Section 3 formulates the loading and unloading strategies and proves their optimality. Section 4 describes the solution method. Section 5 discusses the computational results. Finally, Section 6 gives a conclusion.

2 PROBLEM FORMULATION

2.1 Problem setting

The basic setting of the studied problem is described as follows. For a network with n bike stations, at most $|V|$ vehicles are used for bike repositioning among these stations, where V is the set of vehicles. Each vehicle is assumed to be capacitated and has the same capacity. It is assumed that the number of bikes at every station just before the start of the repositioning activity is known and can be smaller and larger than the demand for bikes there. It is also assumed that there is no change in the number of bikes required at each station during the entire repositioning period. Therefore, bike repositioning is required to take bikes from stations with an excess of bikes to stations with an insufficient number and the number of bikes needed to be loaded or unloaded at each station can be determined before the start of the repositioning activity. Only one depot is considered in this paper, which is referred to as node 0. It is assumed that the depot cannot keep any bike. All vehicles start from the depot simultaneously, travel to their assigned bike stations, and finally return to the depot. Moreover, each station is assumed to be visited once by exactly one vehicle, regardless of the number of vehicles used, to simplify the operation procedure.

The problem determines the optimal routes of all operating vehicles and the loading and unloading quantities at each station to first minimize (1) excess total demand dissatisfaction (i.e., the level of the total demand dissatisfaction of the system above the TDD or the positive deviation from the TDD) with the absolute priority, and then (2) the service time of the operating vehicles. The total demand dissatisfaction of a bike-sharing system is defined as the sum of the number of extra bikes required by each station to serve all users after the end of the repositioning activity. For the service time, in addition to total travel time, this study takes loading and unloading times into consideration. Also, this study considers two service durations. Despite total service time, maximum service duration (in multiple vehicle cases) is considered. This paper, therefore, proposes various mathematical models to determine the route of each of the repositioning vehicles and the loading and unloading quantities at each station.

2.2 Problem formulation

2.2.1 Base model

The notations of this paper are given below:

Sets

N_0	set of nodes, including the depot (which is denoted as 0) and stations;
N	set of nodes (stations), excluding the depot;
V	set of vehicles;
P	set of pickup stations;
R	set of drop-off stations.

Indices

i, j	indices of nodes;
v, v'	indices of vehicles.

Functions

s_i	bike surplus at node i before the repositioning operation starts;
d_i	bike deficit at node i before the repositioning operation starts;
Z	objective function.

Parameters

y_i^I	initial bike inventory level at node i;
y_i^S	target space inventory level at node i, which can be interpreted as the minimum number of empty bike docks reserved to allow all bike users of node i returning bikes to that node or to satisfy the space demand of that node;
y_i^T	target bike inventory level at node i, which can be interpreted as the minimum number of bikes required to serve all bike users of node i borrowing bikes from that node or to serve all bike user demand of that node;
C_i	capacity of node i;
Q	capacity of an operating vehicle;
M	a very large positive constant, i.e., 100,000;
t_{ij}	travel time from node i to node j (excluding loading and unloading times);
L	time needed to remove a bike from a bike station and load it onto a vehicle;
U	time needed to unload a bike from a vehicle and hook it to a locker at a bike station;
TDD	tolerance of total demand dissatisfaction.

Decision variables

x_{ijv}	1 if vehicle v directly travels from node i to node j; 0 otherwise;
q_{ijv}	number of bikes on vehicle v when the vehicle travels directly from node i to node j;
p_{iv}	number of bikes loaded onto vehicle v at node i;
r_{iv}	number of bikes unloaded from vehicle v at node i ;
g_{iv}	auxiliary variable associated with node i and vehicle v used for the sub-tour elimination constraint;
μ	auxiliary variable associated with the excess total demand dissatisfaction of the system.

In this studied BRP, the target space inventory level y_i^S at node i is numerically equal to space demand at that node. The target space inventory level is the minimum number of slots reserved for parking, which is equal to the number of spaces required for bike returns (space demand). The station capacity

C_i is the number of slots provided by station *i*. Both y_i^S and C_i are parameters. Therefore, $C_i - y_i^S$ can be treated as a parameter and interpreted as the "effective capacity" of station *i*, i.e., the number of slots of that station that excludes those reserved for parking. The relative magnitude of $C_i - y_i^S$ and the parameters y_i^T and y_i^I determine the bike surplus and deficit at each station, which are used to define pickup, drop-off, and balanced stations. To deduce bike surplus and deficit of each station, we consider four cases: (1) $C_i - y_i^S \geq y_i^T$ and $y_i^T \geq y_i^I$, (2) $C_i - y_i^S \geq y_i^T$ and $y_i^T < y_i^I$, (3) $C_i - y_i^S < y_i^T$ and $y_i^T \geq y_i^I$, (4) $C_i - y_i^S < y_i^T$ and $y_i^T < y_i^I$.

In Case 1, as $C_i \geq y_i^T + y_i^S$, both bike and space demand can be satisfied simultaneously. Space demand is always satisfied even when no drop-off occurs because $C_i - y_i^S \geq y_i^I$. In this case, $d_i = y_i^T - y_i^I$ to minimize demand dissatisfaction at station *i* and therefore $s_i = 0$ as only either one of d_i and s_i can be positive.

In Case 2, both bike and space demand can be satisfied simultaneously. We further consider two sub-cases: (a) $C_i - y_i^S \geq y_i^I > y_i^T$ and (b) $y_i^I > C_i - y_i^S \geq y_i^T$. For Case (a), space demand is always satisfied even when no pickup occurs at station *i*. The surplus is $y_i^I - y_i^T$ and $d_i = 0$. For Case (b), there are multiple feasible solutions for bike surplus to simultaneously satisfy both bike and space demands. $y_i^I - \left(C_i - y_i^S\right)$ and $y_i^I - y_i^T$ are the lower and upper bounds of the bike surplus, respectively. In this case, we take the upper bound as the bike surplus, i.e., $y_i^I - y_i^T$, to minimize the total demand dissatisfaction of other deficit stations, which is the main consideration of the studied optimization problem. The deficit is, therefore, $d_i = 0$. In both cases, $s_i = y_i^I - y_i^T$ and $d_i = 0$.

In Case 3, both bike and space demand cannot be satisfied simultaneously. Hence, we need to consider which demand satisfaction has a higher priority. When bike demand satisfaction is prioritized over space demand satisfaction, $d_i = y_i^T - y_i^I$ and $s_i = 0$. When space demand satisfaction is prioritized over bike demand satisfaction, we need to consider two more cases: (a) $y_i^I < C_i - y_i^S < y_i^T$ and (b) $C_i - y_i^S < y_i^I \leq y_i^T$. For Case a, $d_i = \left(C_i - y_i^S\right) - y_i^I$ and $s_i = 0$. For Case b, $s_i = y_i^I - \left(C_i - y_i^S\right)$ and $d_i = 0$.

In Case 4, both bike and space demand cannot be satisfied simultaneously. When space demand satisfaction is prioritized over bike demand satisfaction, $s_i = y_i^I - \left(C_i - y_i^S\right)$ and $d_i = 0$. When bike demand satisfaction is prioritized over space demand satisfaction, $s_i = y_i^I - y_i^T$ and $d_i = 0$.

To sum up, $s_i = \max(y_i^I - y_i^T, 0), i \in N$ and $d_i = \max(y_i^T - y_i^I, 0), i \in N$ when bike demand satisfaction is prioritized over space demand satisfaction; $s_i = \max(y_i^I - \left(C_i - y_i^S\right), y_i^I - y_i^T, 0)$, $i \in N$ and $d_i = \max\left(\min\left(C_i - y_i^S, y_i^T\right) - y_i^I, 0\right)$, $i \in N$ when space demand satisfaction is prioritized over bike demand satisfaction.

Based on the above notations and definitions, the base model is formulated as follows:

$$\min Z = \mu \tag{1}$$

subject to

$$\mu \geq 0, \tag{2}$$

$$\mu \geq \sum_{j \in N}\left(d_j - \sum_{v \in V} r_{jv}\right) - TDD, \tag{3}$$

$$p_{jv} - r_{jv} = \sum_{i \in N_0 \backslash \{j\}} q_{jiv} - \sum_{i \in N_0 \backslash \{j\}} q_{ijv}, \qquad \forall j \in N, v \in V; \tag{4}$$

$$p_{jv} \leq s_j, \qquad \forall j \in N, v \in V; \tag{5}$$

$$r_{jv} \leq d_j, \qquad \forall j \in N, v \in V; \tag{6}$$

$$q_{ijv} \leq Qx_{ijv}, \qquad \forall i, j \in N_0, i \neq j, \forall v \in V; \tag{7}$$

$$\sum_{j \in N_0 \backslash \{i\}} x_{ijv} = \sum_{j \in N_0 \backslash \{i\}} x_{jiv}, \qquad \forall i \in N_0, v \in V; \tag{8}$$

$$\sum_{v \in V} \sum_{j \in N_0 \backslash \{i\}} x_{ijv} = 1, \qquad \forall i \in N; \tag{9}$$

$$x_{ijv} = \{0,1\}, \qquad \forall i, j \in N_0, i \neq j, \forall v \in V; \tag{10}$$

$$q_{ijv} \geq 0, \qquad \forall i, j \in N_0, i \neq j, \forall v \in V; \tag{11}$$

$$r_{jv}, p_{jv} \geq 0, \text{integer}, \qquad \forall j \in N, v \in V; \tag{12}$$

$$q_{0iv} = 0, \qquad \forall v \in V, i \in N; \tag{13}$$

$$\sum_{v \in V} x_{0jv} \leq |V|, \qquad \forall j \in N; \tag{14}$$

$$g_{jv} \geq g_{iv} + 1 - M\left(1 - x_{ijv}\right), \qquad \forall v \in V, i \in N_0, j \in N \backslash \{i\}; \tag{15}$$

$$g_{iv} \geq 0, \qquad \forall v \in V, i \in N_0. \tag{16}$$

Equation (1) formulates the objective of the base model, which is to minimize the excess total demand dissatisfaction of the system. Constraints (2) and (3) define the two conditions of the excess total demand dissatisfaction of the system, which can also be combined as $\mu = \max\left(\sum_{j \in N}\left(d_j - \sum_{v \in V} r_{jv}\right) - TDD, 0\right)$. Constraint (4) states that the difference between its loading and unloading quantities at a node for each vehicle equals the difference between its bike load between after and before that node. Constraint (5) requires that the loading quantity of a vehicle at a station cannot exceed the bike surplus of that station. Constraint (6) restricts the unloading quantity of a vehicle at a station not to be greater than the bike shortage of that station. Constraint (7) limits the number of bikes carried on each vehicle to its capacity. Constraint (8) ensures the conservation of vehicle flow. Constraint (9) ensures that each node is visited exactly once by one vehicle only. Constraint (10) defines x_{ijv} to be a binary variable. Constraint (11) defines the number of bikes on any vehicle to be non-negative. Constraint (12) defines the pickup and drop-off quantities of each vehicle at each station to be integral and non-negative. Constraint (13) ensures that all vehicles leaving the depot are empty. Constraint (14) means that not all repositioning vehicles are required to leave the depot. Constraint (15) is the sub-tour elimination constraint. Constraint (16) is the non-negative constraint for the auxiliary variable associated with constraint (15). Note that the integrality of q_{ijv} is implied by that of r_{jv} and p_{jv}.

2.2.2 Model 0

A station i is a pickup station if $s_i > 0$. A station i is a drop-off station if $d_i > 0$. A station i is a balanced station if $s_i = d_i = 0$. For each vehicle, an optimal loading quantity at each pickup station and an optimal unloading quantity at each drop-off station can be respectively expressed as follows:

$$p_{jv} = \min\left(s_j, Q - \sum_{i \in N_0 \setminus \{j\}} q_{ijv}\right), \qquad \forall j \in P, v \in V; \tag{17}$$

$$r_{jv} = \min\left(d_j, \sum_{i \in N_0 \setminus \{j\}} q_{ijv}\right), \qquad \forall j \in R, v \in V. \tag{18}$$

These formulas are for each station and vehicle. However, in the base model, each station is only visited by any vehicle at most once, and hence all routes can be separately considered when determining optimal loading and unloading quantities. This result allows us to simplify the proof of equations (17) and (18) by analyzing optimal loading and unloading strategies route by route. We formally state the result and the proof for each route below:

Proposition 1 For a given route of vehicle $v' \in V$, the following two equations, equations (17a) and (18a), respectively, give an optimal loading quantity at each pickup station and an optimal unloading quantity at each drop-off station that minimizes the total demand dissatisfaction of the route:

$$p_{jv'} = \min\left(s_j, Q - \sum_{i \in N_0 \setminus \{j\}} q_{ijv'}\right), \qquad \forall j \in P \cap J_{v'}; \tag{17a}$$

$$r_{jv'} = \min\left(d_j, \sum_{i \in N_0 \setminus \{j\}} q_{ijv'}\right), \qquad \forall j \in R \cap J_{v'}, \tag{18a}$$

where $J_{v'}$ is the set of nodes for the given route of vehicle $v' \in V$, excluding the depot.

Proof See Appendix A.1.

Equations (17) and (18) only present one of the optimal loading and unloading solutions to the base model and also one of the optimal loading and unloading solutions to the model that minimizes total demand dissatisfaction instead of excess total demand dissatisfaction, because the constant term, TDD, does not affect the optimality. The loading and unloading principle involved is the greedy principle. The loading and unloading quantities determined by these equations are unique. These equations will be used to prove the optimality of the new exact loading and unloading strategies (i.e., equations (41) and (42)) in Proposition 4.

With the newly introduced equations (17) and (18), we can reformulate the base model to become model 0:

$\min Z = \mu$

subject to

constraints (2)–(18).

Model 0 has the same objective as the base model. Meanwhile, it includes two additional constraints, equations (17) and (18), which explicitly define the loading quantity at each pickup station and the unloading quantity at each drop-off station, respectively. These two equations provide cuts for the mathematical model and can be used in exact methods to reduce the computation time for obtaining an optimal solution.

2.2.3 Model 1

The objective of Model 1 is to minimize firstly the excess total demand dissatisfaction of the system, and then the total service time of all repositioning vehicles, in which the latter is the total (in-vehicle) travel, loading, and unloading times. The model is given below:

$$\min Z = \mu \times M + \sum_{v \in V} \left(T_v + S_v \right) \tag{19}$$

subject to
constraints (2)–(18);

$$S_v = \sum_{j \in N} \left(p_{jv} L + r_{jv} U \right), \qquad \forall v \in V; \tag{20}$$

$$T_v = \sum_{i \in N_0} \sum_{j \in N_0 \setminus \{i\}} x_{ijv} t_{ij}, \qquad \forall v \in V. \tag{21}$$

Equation (19) formulates the objective of Model 1. Constraint (20) defines the overall loading and unloading times of a repositioning vehicle along its route, while constraint (21) defines the (in-vehicle) travel time of each vehicle. Note that equations (17) and (18) adopt the greedy loading approach at all stations. This may lead to loading too many bikes on vehicles, and thus some bikes may transport back to the depot and the service time term is not minimized. To deal with that issue, we also modify the pickup strategy in Section 3, where it is analyzed.

2.2.4 Model 2

The objective of Model 2 is to minimize the maximum route duration, which is the maximum sum of total travel, loading, and unloading times of an individual vehicle among all vehicles, after minimizing the excess total demand dissatisfaction of the system. This concept has been employed in the one-commodity pickup-and-delivery vehicle routing problem as a constraint on each route (Shi et al., 2009), and also considered by Schuijbroek et al. (2017) in the objective function of their BRP. This maximum route duration is significant in multiple vehicle cases where the operation duration can be shortened due to more number of vehicles used. Figure 1 gives an example to explain this consideration. Given a small network with one depot D, the minimum total service time is found to be 6 hours when employing one vehicle only, and the time for the repositioning operation is also 6 hours. When considering the minimum repositioning duration, though total service time increases to 7 hours, the operation duration is reduced to 4 hours only. This objective can, therefore, rebalance the system as soon as possible by dividing the workload among vehicles (Schuijbroek et al., 2017) and the duration of the repositioning operation is shortened.

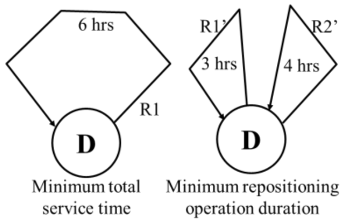


Figure 1 Illustration of the significance of maximum route duration

Considering the importance of maximum service duration, we formulate Model 2 as follows:

$$\min Z = \mu \times M + \Omega \tag{22}$$

subject to

constraints (2)–(18), (20)–(21);

$$\Omega \geq T_v + S_v, \qquad \forall v \in V. \tag{23}$$

Expression (22) shows the objective of Model 2. Similar to Model 1, Model 2 firstly minimizes the excess total demand dissatisfaction of the system, but it subsequently minimizes the maximum route duration Ω among all vehicles as stated in constraint (23). Besides, these models share the same set of constraints.

3 FORMULATION OF THE LOADING AND UNLOADING SUB-PROBLEM

The above bike repositioning problem involves two types of decisions: the routing decision and the loading and unloading decisions. The routing decision is the one about the node visiting sequence of repositioning vehicles while the loading and unloading decisions are those about the number of bikes to be loaded onto the vehicle or unloaded to a bike station along a given route. When the route is given, the bike repositioning problem can be reduced to a loading and unloading sub-problem. Without the second priority level objective as in Model 0, the loading and unloading strategies in Proposition 1 (i.e., the greedy loading approach) are optimal to Model 0. However, with the second priority level objective as in Models 1 and 2, Proposition 1 may not provide optimal solutions to Models 1 and 2 because the surplus bikes loaded on the vehicle increase the service time. Without the use of the greedy loading approach, current approaches to determining optimal loading and unloading strategies (Rainer-Harbach et al., 2015) are time-consuming as they formulate the sub-problem as a linear program or maximum flow problem and solve the sub-problem by commercial solvers in each iteration. This section, therefore, introduces a new set of explicit equations that can determine optimal loading and unloading strategies directly by analyzing the optimality conditions of the sub-problem.

3.1 Terminologies, notations, and sub-problem setting

Before formulating the sub-problem, we define two terminologies: artificial pickup and drop-off nodes. An artificial pickup node is a node formed by one or multiple pickup nodes whereas an artificial drop-off node is a node formed by one or multiple drop-off nodes. The preceding and following nodes of an artificial pickup node are either the depot or artificial drop-off nodes, whereas the preceding and following nodes of an artificial drop-off node are either the depot or artificial pickup-off nodes.

Figure 2 gives an example of a route consisting of three artificial pickup nodes, three artificial drop-off nodes, and two balanced nodes at the two ends of the route (representing the same depot). P_a^v and $P_a'^v$ denote the set of nodes (i.e., balanced and pickup nodes) and the set of pickup nodes that form the a-th artificial pickup node of route v, respectively. R_a^v and $R_a'^v$ denote the set of nodes and the set of drop-off nodes that form the a-th artificial drop-off node of route v, respectively. The number inside a node of an artificial node represents the surplus, balanced, or shortfall, and a positive number means surplus. In the example, the first artificial pickup node consists of two pickup nodes and one balanced node and the first artificial drop-off node consists of only one pick-off node. Based on the definitions of the sets, we obtain $\left|P_1^v\right| = 3$ and $\left|P_1'^v\right| = 2$.

Without loss of generality, we set the artificial node number as the position number of the pair of artificial pickup and drop-off nodes in the route, which means that the *a*-th artificial pickup node is also named artificial pickup node *a*.

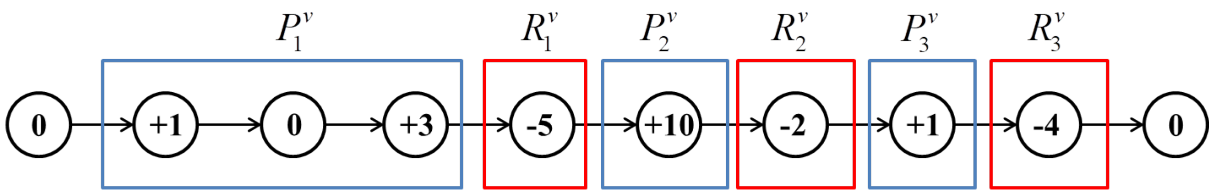


Figure 2 Example of route *v*

The following notations are used in this section.

Sets and indices

P_a^v	Set of nodes (i.e., pickup and balanced nodes) that forms the artificial pickup node *a* of route *v*;
$P_a'^v$	Set of pickup nodes within the artificial pickup node *a* of route *v*;
R_a^v	Set of nodes (i.e., drop-off and balanced nodes) that forms the artificial drop-off node *a* of route *v*;
$R_a'^v$	Set of drop-off nodes within the artificial drop-off node *a* of route *v*;
a,b,c,h,z	Indices for artificial nodes; *a* refers to the depot when it equals 0.
e,f	Indices for nodes that form artificial nodes;
u,v	Indices for vehicles.

Parameters

$s_{a,f}^v$	Number of excess bikes at the *f*-th node in the artificial pickup node *a* of route *v*; $s_{a,f}^v = 0$ if the *f*-th node is a balanced node;
$d_{a,f}^v$	Number of extra bikes required at the *f*-th node in the artificial drop-off node *a* of route *v*; $d_{a,f}^v = 0$ if the *f*-th node is a balanced node;
Q	Capacity of a vehicle (the maximum number of bikes stored on a vehicle);
n^v	Number of artificial pickup nodes in the route of vehicle *v*, which equals the number of artificial drop-off nodes in that route;
M	A large positive number;

Decision variables

p_a^v	Number of bikes picked up at the *a*-th artificial pickup node of route *v*;
r_a^v	Number of bikes dropped off at the *a*-th artificial drop-off node of route *v*.

Functions

$q_{a,a+1}^v$	Number of bikes (i.e., bike load) on vehicle *v* traveled from the *a*-th artificial drop-off node to the (*a*+1)-th artificial pickup node. q_{n^v,n^v+1}^v denotes the bike load from the last artificial drop-off node to the depot;
$p_{a,f}^v$	Number of bikes picked up at the *f*-th node in the *a*-th artificial pickup node of route *v*;

$r_{a,f}^{v}$ Number of bikes dropped off at the f-th node in the a-th artificial drop-off node of route v;

s_{a}^{v} Number of excess bikes at the a-th artificial pickup node of route v (i.e., $\sum_{f=1}^{|P_a^v|} s_{a,f}^{v}$);

d_{a}^{v} Number of extra bikes required at the a-th artificial drop-off node of route v (i.e., $\sum_{f=1}^{|R_a^v|} d_{a,f}^{v}$).

To formulate the sub-problem, artificial nodes in each route are formed. If the first node in the route is a drop-off one, we create a dummy artificial pickup node before the node with $s_1^v = 0$. Analogically, if the final artificial node is the pickup one, we create an artificial drop-off node after the node with $d_{n^v}^{v} = 0$. Meanwhile, for nodes that are balanced initially, they are grouped into the artificial node belonged to the previous node. In other words, if the previous node belongs to the a-th artificial pickup node, the balanced node also belongs to the a-th artificial pickup node. For the exceptional case that the first n nodes are all balanced nodes, they follow the first node in that route that is unbalanced.

Then, without loss of generality, we consider a route with the first artificial node to be an artificial pickup node, the last node to be an artificial drop-off node, and the number of artificial pickup nodes in the route equal to that of artificial drop-off one. Each artificial pickup node is paired up with the next artificial drop-off node. In the example shown in Figure 2, there are three artificial pickup and drop-off pairs. The route starts from and ends at the depot. Each node in the route is visited once. No bike can be kept at the depot so the vehicle leaves and goes back the depot empty. This implies that the total pickup quantity equals the total drop-off quantity for any repositioning vehicle route. It is understood that for each artificial pickup/drop-off node, the total number of bikes to be picked up/dropped off must be a non-negative integer.

3.2 Objective function transformation

Based on inequalities (2) and (3), μ in the objective function can be rewritten as $\max\left(\sum_{j\in N}\left(d_j - \sum_{v\in V} r_{jv}\right) - TDD, 0\right)$. As each station must be visited exactly once by only one vehicle, every bike deficit station is assigned to one of the routes only. Therefore, μ can be further re-written as $\max\left(\sum_{v\in V}\sum_{j\in R\cap J_v}\left(d_j - r_{jv}\right) - TDD, 0\right)$. Moreover, in this sub-problem, as all the routes of the repositioning vehicles are given, the value of x_{ijv}, the set of nodes forming each artificial pickup node P_a^v for each route, and the set of nodes forming each artificial drop-off node R_a^v for each route are all known. Therefore, the total travel time $\sum_{v\in V} T_v$ in equation (21) becomes a constant and the number of artificial pickup nodes (which equals the number of artificial drop-off nodes) of each route is known. Furthermore, because a constant term in an objective function does not affect the optimality conditions, the constant term can be removed. Consequently, objective (19) can be rewritten, in accordance with the new set of notations as

$$\min Z' = \max\left(\sum_{v\in V}\sum_{a=1}^{n^v}\sum_{f=1}^{|R_a^v|}\left(d_{a,f}^v - r_{a,f}^v\right) - TDD, 0\right)\times M + \sum_{v\in V}\sum_{a=1}^{n^v}\left(\sum_{f=1}^{|P_a^v|} p_{a,f}^v L + \sum_{f=1}^{|R_a^v|} r_{a,f}^v U\right), \tag{24}$$

where Z' is the new objective function. The term associated with M in the first part is excess total demand dissatisfaction and the second part calculates the total loading and unloading times. To simplify the expression, let $\pi_{a,f}^v = \sum_{v\in V}\sum_{a=1}^{n^v}\sum_{f=1}^{|R_a^v|}\left(d_{a,f}^v - r_{a,f}^v\right)$.

To minimize Z', the first term in (24), $\max\left(\pi_{a,f}^v - TDD, 0\right)\times M$, should be at least as possible, which implies that the excess total demand dissatisfaction $\max\left(\pi_{a,f}^v - TDD, 0\right)$ should be minimized. Consider two mutually exclusive but collectively exhaustive cases for the excess total demand dissatisfaction: 1) $\pi_{a,f}^v - TDD > 0$, and 2) $\pi_{a,f}^v - TDD \le 0$. In the first case, $\min \max\left(\pi_{a,f}^v - TDD, 0\right) = \min\ \pi_{a,f}^v - TDD$. Given that all $d_{a,j}^v$'s values and TDD are known and fixed, $\min\ \pi_{a,f}^v - TDD$ is equivalent to $\min -\sum_{v\in V}\sum_{a=1}^{n^v}\sum_{f=1}^{|R_a^v|}\left(r_{a,f}^v\right)$. In the second case, $\min\max\left(\pi_{a,f}^v - TDD, 0\right) =$ min 0 = 0, and any feasible value of $r_{a,j}^v$ is optimal in this case. Considering both cases, the minimal excess total demand dissatisfaction is achieved when $-\sum_{v\in V}\sum_{a=1}^{n^v}\sum_{f=1}^{|R_a^v|}\left(r_{a,f}^v\right)$ is minimized. Hence, the objective can be further simplified as

$$\min Z' = -M\sum_{v\in V}\sum_{a=1}^{n^v} r_a^v + \sum_{v\in V}\sum_{a=1}^{n^v}\left(p_a^v L + r_a^v U\right) = \sum_{v\in V}\sum_{a=1}^{n^v}\left(-\left(M+U\right)r_a^v + Lp_a^v\right). \tag{25}$$

As U and L are extremely small when compared with M, they can be excluded from the objective function for simplicity. This gives

$$\min Z' = \sum_{v\in V}\sum_{a=1}^{n^v}\left(-Mr_a^v + p_a^v\right). \tag{26}$$

3.3 The mathematical model of the sub-problem

The mathematical formulation of this sub-problem is given as follows:

$$\min Z' = \sum_{v\in V}\sum_{a=1}^{n^v}\left(-Mr_a^v + p_a^v\right)$$

subject to

$$q_{0,1}^v = 0, \qquad \forall v\in V; \tag{27}$$

$$q_{a,a+1}^v = q_{a-1,a}^v + p_a^v - r_a^v, \qquad \forall a\in\{1,\dots,n^v\}, v\in V; \tag{28}$$

$$s_a^v - p_a^v \ge 0, \qquad \forall a\in\{1,\dots,n^v\}, v\in V; \tag{29}$$

$$d_a^v - r_a^v \ge 0, \qquad \forall a\in\{1,\dots,n^v\}, v\in V; \tag{30}$$

$$Q - \left(p_a^v + q_{a-1,a}^v\right) \ge 0, \qquad \forall a\in\{1,\dots,n^v\}, v\in V; \tag{31}$$

$$q^v_{a,a+1} \geq 0, p^v_a, r^v_a \geq 0, \text{integer} \qquad \forall a \in \{1,\ldots,n^v\}, v \in V. \tag{32}$$

The objective of this sub-problem is to first maximize the total number of bikes dropped off at all artificial drop-off nodes and then minimize the number of bikes to be picked up at all artificial pickup nodes. The constraints of this sub-problem can be directly deduced from constraints (4)-(16), after fixing x_{ijv} values, removing all constraints associated with g_{iv}, and treating each node as an artificial node. Equation (27) (which corresponds to constraint (13)) ensures that the repositioning trucks are empty when leaving the depot. Equation (28) (which corresponds to constraint (4)) is the flow conservation constraint within every pair of artificial pickup and drop-off nodes in a route. Conditions (29) and (30) (which correspond to constraints (5) and (6)) ensure that the numbers of pickup and drop-off bikes at each artificial node should not exceed its bike surplus and shortfall, respectively. Constraint (31) (which corresponds to constraint (7), shown by Proposition 2 below) limits the number of bikes picked up at each artificial pickup node. Condition (32) is the non-negativity and integrality constraint of the decision variables (which corresponds to constraints (11) and (12)).

Proposition 2 Constraint (31) corresponds to constraint (7).

Proof See Appendix A.2.

From (28), the term $q^v_{a,a+1}$ can then be expressed as a function of p^v_a and r^v_a as follows:

$$q^v_{a,a+1} = q^v_{a-1,a} + p^v_a - r^v_a = \left(q^v_{a-2,a-1} + p^v_{a-1} - r^v_{a-1}\right) + p^v_a - r^v_a$$
$$= \left[\left(q^v_{a-3} + p^v_{a-2} - r^v_{a-2}\right) + p^v_{a-1} - r^v_{a-1}\right] + p^v_a - r^v_a = \ldots$$

Therefore,

$$q^v_{a,a+1} = q^v_{0,1} + \sum_{b=0}^{a}\left(p^v_b - r^v_b\right), \qquad \forall a \in \left\{1,\ldots,n^v\right\}, v \in V. \tag{33}$$

Furthermore, it is given that there are no pickup and drop-off at the depot (i.e., $p^v_0 = r^v_0 = 0$) and vehicle v leaves the depot empty (i.e., equation (27)). Equation (33) can, therefore, be rewritten as

$$q^v_{a,a+1} = \sum_{b=1}^{a}\left(p^v_b - r^v_b\right), \qquad \forall a \in \left\{1,\ldots,n^v\right\}, v \in V. \tag{34}$$

Substituting (34) into (31) and (32), the sub-problem can be rewritten with the decision variables of p^v_a and r^v_a only:

$$\min Z' = \sum_{v \in V}\sum_{a=1}^{n^v}\left(-Mr^v_a + p^v_a\right) \tag{35}$$

subject to constraints (29)–(30),

$$Q - \left[p^v_a + \sum_{b=1}^{a-1}\left(p^v_b - r^v_b\right)\right] \geq 0, \qquad \forall a \in \left\{1,\ldots,n^v\right\}, v \in V; \tag{36}$$

$$\sum_{b=1}^{a}\left(p^v_b - r^v_b\right) \geq 0, \qquad \forall a \in \left\{1,\ldots,n^v\right\}, v \in V; \tag{37}$$

$$p^v_a, r^v_a \geq 0, \text{integer}, \qquad \forall a \in \left\{1,\ldots,n^v\right\}, v \in V. \tag{38}$$

For any given route of vehicle v, the optimal loading and unloading strategies of Proposition 1, denoted as $p_a^{v,1}$ and $r_a^{v,1}$ correspondingly, can be expressed in terms of the new set of notations: $p_a^{v,1}=\min\left(s_a^v, Q-\sum_{b=1}^{a-1}\left(p_b^{v,1}-r_b^{v,1}\right)\right)$ and $r_a^{v,1}=\min\left(d_a^v, p_a^{v,1}+\sum_{b=1}^{a-1}\left(p_b^{v,1}-r_b^{v,1}\right)\right)$, for all $a\in\{1,\ldots,n^v\}$. To simplify the expressions, we let $\Delta_a^{v,1}=\sum_{b=0}^{a}\left(p_b^{v,1}-r_b^{v,1}\right)$. Then, by definition, we have $\Delta_0^{v,1}=0$, $v\in V$,

$$p_a^{v,1}=\min\left(s_a^v, Q-\Delta_{a-1}^{v,1}\right),\ \forall a\in\{1,\ldots,n^v\}, v\in V, \text{ and} \tag{39}$$

$$r_a^{v,1}=\min\left(d_a^v, p_a^{v,1}+\Delta_{a-1}^{v,1}\right),\ \forall a\in\{1,\ldots,n^v\}, v\in V. \tag{40}$$

3.4 New loading and unloading strategies at each artificial node in a route

3.4.1 Zero tolerance of total demand dissatisfaction

The loading and unloading strategies of Proposition 1 can minimize the demand dissatisfaction of a given route, but they may require the vehicle to load excess bikes and therefore the resultant service time may not be minimal. To minimize the service time of the route and attain the minimum demand dissatisfaction, a new loading strategy is given as

$$p_a^{v,2}=\min\left(s_a^v, Q-\Delta_{a-1}^{v,2}, d_a^v+\sum_{b=a+1}^{n^v}\left\{\max\left[d_b^v-\min\left(s_b^v,Q\right),0\right]\right\}-\Delta_{a-1}^{v,2}\right),\ \forall a\in\{1,\ldots,n^v\}, v\in V, \tag{41}$$

and the unloading strategy $r_a^{v,2}$ is

$$r_a^{v,2}=\min\left(d_a^v, p_a^{v,2}+\Delta_{a-1}^{v,2}\right),\ \forall a\in\{1,\ldots,n^v\}, v\in V, \tag{42}$$

where $\Delta_a^{v,2}$ is the bike load on vehicle v traveling from the a-th artificial drop-off node to the (a+1)-th artificial pickup node based on these new strategies, and hence

$$\Delta_a^{v,2}=\sum_{b=0}^{a}\left(p_b^{v,2}-r_b^{v,2}\right) \text{ and } \Delta_0^{v,2}=0. \tag{43}$$

The first and second terms in the minimum operator of condition (41) are equivalent to those in the optimal loading strategy in Proposition 1, which represent the cases of loading all bikes at each artificial pickup node on the vehicle and loading bikes until the vehicle is full, respectively. In the third term, the expression $\max\left[d_b^v-\min\left(s_b^v,Q\right),0\right]$ represents the number of bikes required by artificial drop-off node b subsequent to artificial pickup node a after considering the bike surplus at artificial pickup node b and vehicle capacity. The term $\sum_{b=a+1}^{n^v}\left\{\max\left[d_b^v-\min\left(s_b^v,Q\right),0\right]\right\}$ captures the total number of outstanding bikes at all artificial drop-off nodes subsequent to artificial pickup node a, after considering all bike surpluses in the subsequent artificial pickup nodes and vehicle capacity. The third term represents the minimum number of required bikes to satisfy both the bike requirement at the a-th artificial drop-off node and the extra bike requirements at subsequent artificial nodes. To further simplify the expression, let $\phi_{a+1}^v=\sum_{b=a+1}^{n^v}\left\{\max\left[d_b^v-\min\left(s_b^v,Q\right),0\right]\right\}$, or

$$\phi_{a+1}^{v} = \sum_{b=a+1}^{n^{v}} \max\left(d_{b}^{v} - s_{b}^{v}, d_{b}^{v} - Q, 0\right). \tag{44}$$

It is also noted that $\phi_{a+1}^{v} \geq 0$.

Table 2 compares the results obtaining by implementing the new loading and unloading strategies (41)-(42) and the old ones (39)-(40) and using the example in Figure 2. The major difference is the operation at the second artificial pickup node, in which the new strategy (41)-(42) results in picking up fewer bikes at that node than the old strategy. Given that the vehicle capacity Q equals 20, the pickup quantity at the second artificial pickup node according to condition (39) is $\min\left(s_{2}^{v}, Q - \Delta_{1}^{v,1}\right) = \min\left(10, 20\right) = 10$. This implies that the vehicle needs to pick up all surplus bikes at the second artificial pickup node. From (41), we have $\phi_{3}^{v} = \sum_{b=3}^{3} \max\left(d_{b}^{v} - s_{b}^{v}, d_{b}^{v} - Q, 0\right) = \max\left(3, -16, 0\right) = 3$. It means that the subsequent artificial drop-off node requires three extra bikes from the preceding artificial pickup nodes to satisfy its demand. The pickup quantity at the second artificial pickup node based on condition (41) becomes $\min\left(s_{2}^{v}, Q - \Delta_{1}^{v,2}, d_{2}^{v} + \phi_{3}^{v} - \Delta_{1}^{v,2}\right) = \min\left(10, 20, 5\right) = 5$. Table 2 also shows that this reduction in loading activities does not necessarily increase the demand dissatisfaction at subsequent artificial nodes.

Table 2 Comparison of the results obtained by implementing the loading and unloading strategies (39)-(40) and (41)-(42) and using the example in Figure 2

Strategy	$s_1^v(p_1^v)$	$d_1^v(r_1^v)$	Δ_1^v	dd_1^v	$s_2^v(p_2^v)$	$d_2^v(r_2^v)$	Δ_2^v	dd_2^v	$s_3^v(p_3^v)$	$d_2^v(r_2^v)$	Δ_3^v	dd_3^v
(39)-(40)	4 (4)	5 (4)	0	1	10 (10)	2 (2)	8	0	1 (1)	4 (4)	5	0
(41)-(42)	4 (4)	5 (4)	0	1	10 (5)	2 (2)	3	0	1 (1)	4 (4)	0	0

Note: dd_a^v denotes the demand dissatisfaction at artificial drop-off node a in the route of vehicle v, which equals $d_a^v - r_a^v$.

Proposition 3 For a given route of truck v, the loading strategy (41) and the unloading strategy (42) give a feasible loading quantity at each artificial pickup node and a feasible unloading quantity at each artificial drop-off node, respectively, with respect to constraints (29)–(30) and (36)–(38).

Proof See Appendix A.3.

Proposition 4 For a given route of truck v, the loading strategy (41) and the unloading strategy (42) achieve the equivalent total demand dissatisfaction as those strategies stated in Proposition 1, given that the tolerance of total demand dissatisfaction (TDD) equals 0.

Proof See Appendix A.4.

Proposition 5 For a given route of truck v and TDD = 0, the loading strategy (41) and the unloading strategy (42) give an optimal loading quantity at each artificial pickup node and an optimal unloading quantity at each artificial drop-off node, respectively.

Proof This follows directly from Proposition 3 and Proposition 4.

3.4.2 Positive tolerance of total demand dissatisfaction

Let p_a^{v*} and r_a^{v*} be the optimal loading and unloading quantities at artificial pickup and drop-off node a in the route of vehicle v obtained by solving equations (41) and (42), respectively. When $TDD > 0$, these quantities may or may not be optimal. We consider two cases:

Case 1: $\sum_{v \in V} \sum_{a=1}^{n^v} \left(d_a^v - r_a^{v*} \right) \geq TDD$, and

Case 2: $\sum_{v \in V} \sum_{a=1}^{n^v} \left(d_a^v - r_a^{v*} \right) < TDD$.

In Case 1, the optimal loading and unloading quantities at each artificial node for $TDD = 0$ are also optimal for $TDD > 0$. The loading strategy (41) and the unloading strategy (42) can also be used as the optimal loading and unloading strategies for $TDD > 0$.

In Case 2, the optimal unloading strategy for $TDD = 0$, $r_a^{v*}, \forall a$, leads to the total demand dissatisfaction of the system to be lower than TDD. This implies that there is an updated optimal unloading quantity $r_a^{v,opt}$ at some artificial drop-off node a such that $r_a^{v,opt}$ is smaller than r_a^{v*} and $\sum_{v \in V} \sum_{a=1}^{n^v} \left(d_a^v - r_a^{v,opt} \right) = TDD$. The loading strategy (41) and the unloading strategy (42) can be modified to determine the loading and unloading quantities of each artificial node.

For Case 2, we know that $r_a^{v,opt} \leq r_a^{v*}, \forall a, v$ in general and $r_a^{v,opt} = r_a^{v*}$ for some artificial drop-off nodes. There are multiple combinations for choosing some out of a total of $n^v |V|$ drop-off quantity variables so that $r_a^{v,opt}$ is set to be less than r_a^{v*}(i.e., $r_a^{v,opt} < r_a^{v*}$) for a chosen variable, $r_a^{v,opt} = r_a^{v*}$ for a non-chosen variable, and $\sum_{v \in V} \sum_{a=1}^{n^v} \left(d_a^v - r_a^{v,opt} \right) = TDD > 0$. When the difference $TDD - \sum_{v \in V} \sum_{a=1}^{n^v} \left(d_a^v - r_a^{v*} \right) = \omega \leq n^v |V|$, there are ${}_{n^v|V|}C_{\omega - w'}$ ways to choose the variables so that $r_a^{v,opt}$ is less than r_a^{v*} by a maximum difference of $w' + 1$, where $0 \leq w' \leq \omega - 1$. Moreover, the maximum possible number of strict inequalities, $\omega_{\max}$, equals $\min\left(TDD - \sum_{v \in V} \sum_{a=1}^{n^v} \left(d_a^v - r_a^{v*} \right), n^v |V| \right)$. Therefore, the number of optimal strategies equals $\sum_{\omega=1}^{\omega_{\max}} {}_{n^v|V|}C_{\omega}$. This number does not consider the fact that when there are two or more artificial drop-off quantity variables chosen, there can be multiple solutions satisfying $\sum_{v \in V} \sum_{a=1}^{n^v} \left(d_a^v - r_a^{v,opt} \right) = TDD$. Hence, the number of optimal strategies at least equals $\sum_{\omega=1}^{\omega_{\max}} {}_{n^v|V|}C_{\omega}$. To summarize, we give the following.

Proposition 6 When $\sum_{v \in V} \sum_{a=1}^{n^v} \left(d_a^v - r_a^{v*} \right) < TDD$, at least $\sum_{\omega=1}^{\omega_{\max}} {}_{n^v|V|}C_{\omega}$ optimal unloading strategies can be modified from the unloading strategy (42), where $\omega_{\max} = \min\left(TDD - \sum_{v\in V}\sum_{a=1}^{n^v}\left(d_a^v - r_a^{v*}\right), n^v|V|\right)$.

It is noted that smaller unloading quantities at the artificial drop-off nodes with $r_a^{v,opt} < r_a^{v*}$ in Case 2 implies smaller loading quantities at the preceding artificial pickup nodes of the corresponding routes, both of which contribute to smaller total service time compared with Case 1. Moreover, smaller loading quantities imply that multiple optimal loading strategies can be modified from the loading strategy (41).

3.5 Loading/unloading strategy within artificial nodes

3.5.1 Zero tolerance of total demand dissatisfaction

Proposition 5 only determines the number of bikes to be picked up at an artificial pickup node p_a^v. For an artificial pickup node with more than one pickup node, if the bike surplus of the artificial node is larger than the number of bikes needed to be loaded (i.e., $s_a^v > p_a^v$), there exist multiple optimal pickup strategies within the artificial node. To explain this, we consider an artificial pickup node with $\left|P_a'^v\right| = t$ ($t > 1$) pickup nodes with $s_{a,f}^v > 0$. There is a total of t pickup decisions, which are constrained by t upper bound constraints for pickup quantities (i.e., $p_{a,f}^v \le s_{a,f}^v$) and the equality constraint for the pickup requirement (i.e., $\sum_{f=1}^{|P_a^v|} p_{a,f}^v = p_a^v$). Because $s_a^v > p_a^v$, to ensure the equality condition to be satisfied, at least one upper bound constraint must not be binding. In general, there are ${}_tC_\sigma$ optimal pickup strategies if there are σ non-binding upper bound constraints. Moreover, the maximum number of non-binding upper bound constraints $\sigma_{\max}$ equals $\min\left(s_a^v - p_a^v, t\right)$. The number of optimal pickup strategies therefore equals $\sum_{\sigma=1}^{\sigma_{\max}} {}_tC_\sigma$. This number does not take into account the fact that when there are two pickup nodes with $p_{a,f}^v < s_{a,f}^v$, there can be multiple solutions of $p_{a,f}^v$ for these nodes to achieve $\sum_{f=1}^{|P_a^v|} p_{a,f}^v = p_a^v$. Hence, we can state the following.

Proposition 7 When at least one artificial pickup node has t pickup nodes and the bike surplus of the artificial node is larger than the number of bikes needed to be loaded from the artificial node (i.e., $s_a^v > p_a^v$), there are at least $\sum_{\sigma=1}^{\sigma_{\max}} {}_tC_\sigma$ optimal loading strategies within the artificial node, where $\sigma_{\max} = \min(s_a^v - p_a^v, t)$.

Analogically, we consider an artificial drop-off node having more than one drop-off node, $d_a^v > r_a^v$, the equality condition for the drop-off strategy $\sum_{f=1}^{|R_a^v|} r_{a,f}^v = r_a^v$, and $r_{a,f}^v \le d_{a,f}^v$. The analysis of multiple pickup strategies within an artificial pickup node is also applicable to show the existence of multiple optimal drop-off strategies within an artificial drop-off node. Hence, we can state the following.

Proposition 8 When at least one artificial drop-off node has t drop-off nodes and the bike deficit of the artificial node is larger than the number of bikes available to unload at the artificial node (i.e., $d_a^v > r_a^v$), there are at least $\sum_{\sigma=1}^{\sigma_{\max}} {}_tC_\sigma$ optimal unloading strategies within the artificial node, where $\sigma_{\max} = \min(d_a^v - r_a^v, t)$.

From Proposition 7 and Proposition 8, it can be seen that multiple optimal loading and unloading strategies exist within an artificial node. For simple calculation, this study further assumes that each vehicle loads all excess bikes of each pickup node in an artificial pickup node, starting from the first pickup node until either all excess bikes at each pickup node in the artificial pickup node are loaded to the vehicle concerned or the target number of bikes to be loaded from the artificial pickup node has been reached. Let $B_{a,f}^v$ be the remaining number of bikes needed to be picked up after passing through the (f-1)-th node of the artificial pickup node a in the route of vehicle v in order to reach a total number of p_a^{v*} for TDD = 0. An optimal pickup strategy for each node becomes $p_{a,f}^v = \min\left(s_{a,f}^v, B_{a,f}^v\right)$, where $B_{a,f}^v = \max\left(p_a^{v*} - \sum_{f'=1}^{f-1} s_{a,f'}^v, 0\right)$, which is equivalent to

$$p_{a,f}^v = \min\left(s_{a,f}^v, \max\left(p_a^{v*} - \sum_{f'=1}^{f-1} s_{a,f'}^v, 0\right)\right), \forall a \in \{1,\ldots,n^v\}, v \in V, f \in \{1,\ldots,|P_a^v|\}. \tag{45}$$

Note that vehicle capacity is not considered in this expression because p_a^{v*} is smaller than that implicitly.

Proposition 9 Condition (45) represents an optimal pickup strategy for each node forming the a-th artificial pickup node with $\left|P_a'^v\right| \geq 1$ in the route of vehicle v when TDD = 0.

Proof See Appendix A.5.

Similarly, it is assumed that each vehicle unloads as many bikes as possible at each drop-off node in an artificial node starting from the first drop-off node, and stops unloading when either the deficit of each drop-off node is satisfied or the target number of bikes to be dropped off is reached. The optimal unloading strategy for each drop-off node is

$$r_{a,f}^v = \min\left(d_{a,f}^v, \max\left(r_a^{v*} - \sum_{f'=1}^{f-1} d_{a,f'}^v, 0\right)\right), \ \forall a \in \{1,\ldots,n^v\}, v \in V, f \in \{1,\ldots,|R_a^v|\}. \tag{46}$$

The second term of the minimum operator for $r_{a,f}^v$ represents the number of bikes remained on vehicle v after leaving the (f-1)-th node in artificial drop-off node a, and is smaller than the first term $d_{a,f}^v$ when the number of bikes on vehicle v is insufficient to satisfy the deficit at the f-th node in artificial drop-off node a.

Proposition 10 Equation (46) represents an optimal drop-off strategy for each node forming the a-th artificial drop-off node with $\left|R_a'^v\right| \geq 1$ in the route of vehicle v when TDD = 0.

Proof See Appendix A.6.

3.5.2 Positive tolerance of total demand dissatisfaction

As implied by the discussion in Section 3.4.2, when $TDD > 0$, the total drop-off quantity at all artificial nodes is smaller than that when $TDD = 0$. We reduce the drop-off quantity at each artificial node one by one until the total reduction is equal to TDD. The adjustment follows first the order of artificial node position in a route and then vehicle number. The updated optimal unloading strategy for the a-th artificial drop-off node in the route of vehicle v', $r_a^{v',opt}$, becomes

$$r_a^{v',opt} = \min\left(r_a^{v'*}, \max\left(0, \sum_{v=1}^{v'-1}\sum_{c=1}^{n^{v'}} r_c^{v'*} + \sum_{h=1}^{a} r_h^{v'*} - TDD \right)\right), \ \forall a \in \{1,\ldots,n^{v'}\}, v' \in V. \tag{47}$$

Proposition 11 Condition (47) represents an optimal drop-off strategy for each artificial drop-off node in each route when TDD is positive.

Proof See Appendix A.7.

By replacing v in (46) with v' and then $r_a^{v'*}$ with $r_a^{v',opt}$, an optimal drop-off strategy at the f-th node within artificial drop-off node a with $\left|R_a'^v\right| \geq 1$ in the route of vehicle v' follows a similar strategy to strategy (46), i.e., $r_{a,f}^{v'} = \min\left(d_{a,f}^{v'}, \max\left(r_a^{v',opt} - \sum_{f'=1}^{f-1} d_{a,f'}^{v'}, 0 \right)\right)$. (48)

Let $TDD_{v'}$ be the TDD of the route of vehicle v' such that $TDD^{v'} = \sum_{c=1}^{n^{v'}}\left(r_c^{v'*} - r_c^{v',opt} \right)$ and $TDD = \sum_{v=1}^{|V|} TDD^v$. The updated loading strategy at the a-th artificial pickup node in the route of vehicle v', $p_a^{v',opt}$, is given by

$$p_a^{v',opt} = \min\left[p_a^{v'*}, \max\left(0, \sum_{c=1}^{a} p_c^{v'*} - TDD^{v'} \right)\right], \ \forall a \in \{1,\ldots,n^{v'}\}, v' \in V. \tag{49}$$

Proposition 12 Equation (49) represents an optimal pickup strategy for each artificial pickup node a in the route of vehicle v' when TDD is positive.

Proof See Appendix A.8.

Replacing p_a^{v*} in (45) with $p_a^{v,opt}$, an optimal pickup strategy at the f-th node of artificial pickup node a with at least one pickup node in the route of vehicle v can be given as

$$p_{a,f}^{v} = \min\left(s_{a,f}^{v}, \max\left(p_a^{v,opt} - \sum_{f'=1}^{f-1} s_{a,f'}^{v}, 0 \right)\right). \tag{50}$$

4 SOLUTION METHOD

To obtain solutions to the proposed models, the existing solution methods for BRPs as shown in Table 3 can be used. As reflected from this table, exact methods, such as branch-and-cut algorithms (see Dell'Amico et al., 2014; Erdoğan et al., 2014, 2015), have been used to solve BRPs. However, it is intractable to use exact methods to solve a large, realistic BRP because the problem is NP-hard. Previous studies (e.g., Raviv et al., 2013; Ho and Szeto, 2014) have also illustrated this point by conducting numerical experiments. Hence, most of the existing studies focus on the development of inexact methods, including heuristics, metaheuristics, and hybrids of exact methods and (meta)heuristics, to obtain good solutions to BRPs with short computing times. Therefore, this study also aims to develop an efficient inexact method to solve a large, realistic BRP with short computation time, instead of solving the proposed models directly by exact methods. Nevertheless, the optimal solutions derived from the analysis of the sub-problem in Section 3 will be used in the proposed solution method to improve the solution quality.

Table 3 Summary of the solution methods for BRPs

Approach	Solution Method	References
Exact	Branch-and-cut algorithm	Dell'Amico et al. (2014); Erdoğan et al. (2014, 2015)
Approximation	9.5-approximation algorithm	Benchimol et al. (2011)
Heuristics/metaheuristics	Large neighborhood search (LNS)	Di Gaspero et al. (2013b)
	Iterated tabu search	Ho and Szeto (2014)
	PILOT/GRASP + variable neighborhood descent	Kloimüllner et al. (2014); Rainer-Harbach et al. (2015)
	Variable neighborhood search (VNS)	Raidl et al. (2013); Kloimüllner et al. (2014); Rainer-Harbach et al. (2015)
	Memetic algorithm	Ting and Liao (2013)
	Insertion heuristic	Alvarez-Valdes et al. (2016)
	Hybrid iterated local search	Cruz et al. (2017)
	Destroy and repair algorithm	Dell'Amico et al. (2016)
	Chemical reaction optimization	Szeto et al. (2016)
	Hybrid genetic algorithm	Li et al. (2016)
	Cluster-first route-second heuristic	Schuijbroek et al. (2017)
	Hybrid large neighborhood search	Ho and Szeto (2017)
Hybrids of exact methods and (meta)heuristics	Branch-and-cut algorithm with tabu search	Chemla et al. (2013)
	Multistage heuristic incorporating CPLEX	Angeloudis et al. (2014)
	3-step math heuristic	Forma et al. (2015)
	Branch-and-bound algorithm with Genetic Algorithm (GA)/ Greedy Search algorithm/ Nearest neighbor algorithm	Kadri et al. (2016)

As seen from Table 3, only limited types of heuristics are adopted to solve BRPs. In this study, an enhanced Artificial Bee Colony (ABC) algorithm is adopted as the main solution method to solve the problem. This algorithm is an improved version of the ABC algorithm firstly proposed by Karaboga (2005). The ABC algorithm is a swarm-based metaheuristic that mimics the intelligent behavior of the honeybees' foraging process to solve optimization problems. In the ABC algorithm, the bees are divided into three types: employed bees, onlookers, and scouts. Employed bees are responsible for exploiting available food sources and gathering required information. They also share the information with the onlookers, and the onlookers select existing food sources to be further explored. When the food source is abandoned by its employed bee, the employed bee becomes a scout and starts to search for a new food source in the vicinity of the hive. The abandonment happens when the quality of the food source is not improved after performing a maximum allowable number of iterations. For the details of the ABC algorithm, the readers can refer to the studies of Karaboga (2005; 2008).

Like classical metaheuristics such as GA, the ABC algorithm is population-based. However, in contrast to the local search ability of GA, the ABC algorithm possesses better local search ability due to the use of two independent types of bees called employed and onlooker bees. Previous studies also show that the ABC algorithm can outperform GA (e.g., Karaboga and Basturk, 2007; Szeto and Jiang, 2014). The ABC algorithm has an advantage of easy-to-use. It is not bounded by the mathematical properties of the objectives and constraints; it can find nearly optimal solutions with much shorter computational time compared with CPLEX (see Section 5.2) or other existing heuristics (see Szeto et al., 2011). It has been demonstrated that the ABC algorithm can be applied to solve various discrete and combinatorial optimization problems with great success. Readers can refer to the study of Karaboga et al. (2014) for some examples of the successful applications of the ABC algorithm and its variants towards those problems.

Among the modified versions of ABC algorithms, Szeto et al. (2011) constructed a modified version of the ABC algorithm, called the enhanced ABC (EABC) algorithm, which has shown to be more effective than the ABC algorithm and some of the classic methods to solve the capacitated vehicle routing problem (CVRP). Because the routing problem in this study is similar to the CVRP in that study, this study applies the EABC algorithm to determine vehicle routes and uses exact loading and unloading strategies depicted in the last section to determine pickup and delivery quantities at each station in each given route in each solution obtained from the EABC algorithm. Meanwhile, it is noted that the EABC algorithm can be replaced by other metaheuristics (as shown in Table 3), which can undergo route search.

The main steps of the EABC algorithm are summarized below, where the boldface denotes the new changes made by Szeto et al. (2011):

1. Initialize *limit*, the maximum number of cycles (i.e., the maximum number of iterations), and the number of food sources (i.e., the number of solutions) Y; Initialize $W = 0$ and $H_1 = H_2 = \ldots = H_Y = 0$.
   where W = Number of times of repeating a whole foraging process;
   H_y = Number of times of applying a neighborhood operator to food source y;
   $y = 1, \ldots, Y$;
2. Y food sources (w_y, $y = 1, \ldots, Y$) are randomly generated.
3. Each employed bee is assigned to a food source. Each food source's objective value $Z(w_y)$ and fitness value $F(w_y)$ are evaluated.
4. If W = maximum number of cycles, stops.
5. Do the following foraging process:
   a. Employed Bee Phase
      i. A neighborhood operator is applied to each food source: $w_y \rightarrow \dot{w}_y$, $y = 1, \ldots, Y$.
      ii. For each food source w_y,
         if $F(\dot{w}_y) > F(w_y)$, w_y is replaced by $\dot{w}_y$ and $H_y = 0$; otherwise, $H_y = H_y + 1$.
   b. Onlooker Bee Phase

i. Each onlooker bee selects a food source using the roulette wheel selection method.
ii. A neighborhood operator is applied to each selected food source: $w_y \rightarrow \ddot{w}_y$, $y = 1, \ldots, Y$.
iii. For each new food source $\ddot{w}_y$,

if $F(\ddot{w}_y) > F(w_y)$, select $w_{y'}$ where $H_{y'}$ is the maximum among all food sources;
If $F(\ddot{w}_y) > F(w_{y'})$, $w_{y'}$ is replaced by $\ddot{w}_y$ and $H_{y'} = 0$; otherwise, $H_{y'} = H_{y'} + 1$.

c. Scout Bee Phase
i. For each food source w_y,

if $H_y = limit$, **a neighborhood operator is applied to the food source: $w_y \rightarrow \dot{w}_y$ and w_y is replaced by $\dot{w}_y$.**

6. $W = W + 1$; go to Step 4.

4.1 Solution representation

The solution is represented by a sequence of nodes (i.e., stations and the depot). Each station is given an index number, e.g., 1 to 10 for a network with 10 stations. The depot is represented by 0. The number of '0's used in a sequence equals the number of vehicles used in that solution minus one. For example, if a solution has 10 stations that are visited by 3 vehicles, the representation is

$$0 \rightarrow 1 \rightarrow 4 \rightarrow 5 \rightarrow 0 \rightarrow 2 \rightarrow 6 \rightarrow 3 \rightarrow 0 \rightarrow 8 \rightarrow 10 \rightarrow 7 \rightarrow 9 \rightarrow 0$$

There are four zeros, meaning that there are three vehicles. The first vehicle departs from the depot, passes through stations 1, 4, and 5 subsequently, and then travels back to the depot. The second vehicle departs from the depot, passes through stations 2, 6, and 3 subsequently, and then travels back to the depot. The third vehicle departs from the depot, passes through stations 8, 10, 7, and 9 subsequently, and then travels back to the depot. Note that we just use one zero instead of two zeros to separate two adjacent routes in the middle of the solution to save computer memory and all vehicles actually depart from the depot simultaneously.

4.2 Neighborhood operators

Three neighborhood operators—random swapping, reversing a subsequence, and random swapping two reversed subsequences—are used to alter the positions of different bike stations in a node sequence so that a new solution can be obtained from an old solution. These three operators can be employed solely to generate new solutions and the mechanisms of these operators can refer to the study of Szeto et al. (2011). Meanwhile, through the computational experiments in our preliminary study, it was found that a new neighborhood operator, which is formed by randomly choosing one of these three operators in each neighborhood search, is more effective than employing either one neighborhood operator alone in solution search.

4.3 Fitness value determination

The fitness value of a food source w_y is computed by $F(w_y) = 1/Z(w_y)$. Note that when calculating $Z(w_y)$, the in-vehicle travel times and the loading and unloading times of the route and the loading and unloading quantities at each station are required. The in-vehicle travel times and the loading and unloading times of all routes are calculated on the basis of the principle of (21) and (20) respectively, and the loading and unloading quantities are calculated on the basis of the principle of (41)-(43), (45)-

(50) for each given w_y.

4.4 Roulette wheel selection method

During each onlooker bee phase, a food source w_y is randomly selected by a roulette wheel selection method. The probability of choosing that food source, $S(w_y)$, is calculated as $S(w_y) = F(w_y) / \sum_{y=1}^{Y} F(w_y)$, $y = 1, \ldots, Y$.

5 COMPUTATIONAL RESULTS

All experiments were carried out on a computer equipped with Windows 7, an Intel(R) Core(TM) Duo CPU E8500@ 3.16GHz, and a 4.00GB of RAM, and the program was coded using Dev-C++ 4.9.9.2. For parameter setting, $Y = 25$, which is also the number of employed bees and the number of onlooker bees. This gives a bee colony size (i.e., the total number of employed bee and onlookers) of 50. This is based on the study of Karaboga and Basturk (2008), in which the number of employed bees was set to be equal to the number of onlookers to reduce the number of parameters, and their study found that the colony size of 50 could provide an acceptable convergence speed for search.

5.1 Test Instances

The EABC algorithm was used to solve the test instances introduced in Chemla et al. (2013), where the instances have the number of stations $|N| = n \in \{30, 60, 99\}$. The algorithm was also used to solve the test instances with larger network sizes as introduced in Rainer-Harbach et al. (2013) with $|N| = n \in \{180, 300, 500, 700\}$. The number of vehicles used varies from 1 to 5. For each vertex i in the network, either s_i or d_i does not exceed 20, which is the capacity of one repositioning vehicle. Both the loading time L and the unloading time U of each bike are set to be 30 seconds, and each vehicle is assumed to travel at a low and uniform speed of 4 m/s (i.e., 14.4 km/h) to mimic the worst case situation. Without loss of generality, we assume that the target bike inventory level is equal to the difference between the station capacity and the target space inventory level. For notation purposes, we define 1v60n as the instance with one available vehicle and a 60-station network. Other instances are defined similarly.

5.2 A comparison of the performance between the EABC algorithm and CPLEX

To assess the performance of the EABC algorithm, the results of the EABC algorithm were compared with those of CPLEX. Both methods were assessed by 28 instances. Other than the 12 instances generated from $|N| = n \in \{30, 60, 99\}$ based on instances from Chemla et al. (2013) with $|V| = \{1, 2, 3, 4\}$, this assessment also used the instances from Rainer-Harbach et al. (2013) with $|N| = n \in \{180, 300, 500, 700\}$ and $|V| = \{1, 2, 3, 4\}$ to compare the efficiency of the EABC algorithm with CPLEX in solving large networks. In all test instances, they were tested under the objective of Model 1. To reduce the running time of CPLEX, the model has been reformulated using new loading and unloading decision variables and a few valid inequalities, proposed by Raviv et al. (2013), have been added to the revised model. The revised version of Model 1 and the details of these valid inequalities can be referred to Appendix D.

Table 4 shows the running times (CPU) in seconds, upper bounds (UBs), lower bounds (LBs), and gaps obtained by CPLEX, in which CPLEX stopped after reached a 2-hour limit or an optimal solution was found. Table 4 also shows the average computation time (s) and the average and the best objective values obtained by the EABC algorithm in 20 runs as well as the gap (%) representing the deviation of the average objective value from the corresponding LB. When $|N| = 30$ and $|V| = 1$, CPLEX could get an optimal solution in about 15 seconds while the EABC algorithm took less than 7 seconds to get the same solution. For slightly larger fleet sizes ($|V| = 2, 3$), optimal solutions could be obtained by CPLEX using a much longer running time where optimal solutions could be obtained by the EABC algorithm using about the same time as the case of $|V| = 1$. When the fleet size further increases (i.e., $|V| = 4$), only a feasible solution with a gap of 2.2% could be obtained by CPLEX. Meanwhile, the solution obtained by the EABC algorithm is better than the solution obtained by CPLEX in this case.

Table 4 Comparison of experimental results between the EABC algorithm and CPLEX

Instances	EABC				CPLEX			
	Average[a]	Minimum[b]	Gap%[c]	CPU (s)	UB	LB	Gap%[d]	CPU (s)
1v30n	6106.5	6106.5	0	6.5	6106.5	6106.5	0	14.8
1v60n	11809.5	11743	1.3	19.8	11658.5	11658.5	0	2100.2
1v99n	23228.6	23072.5	3.7	52.4	26231.5	22394.7	14.6	7200
1v180n	37502.8	37495.3	0.1	78.3	176698.8	37459.1	78.8	7200
1v300n	61564.2	61555.5	1.3	141.6	3122830.3	60795.6	98.1	7200
1v500n	97039.7	97012.3	-	259.2	-	-	-	7200
1v700n	140489.3	140466.3	-	484.8	-	-	-	7200
2v30n	6106.5	6106.5	0	6.6	6106.5	6106.5	0	512
2v60n	11821.2	11684.5	1.4	20.5	11658.5	11658.5	0	4315.2
2v99n	23362.4	23072.5	4.7	51.2	-	22321.2	-	7200
2v180n	37502.8	37497	0.1	79	-	37458.2	-	7200
2v300n	61563.9	61554.5	-	142.7	-	-	-	7200
2v500n	97040.9	97016.8	-	259.9	-	-	-	7200
2v700n	140486.6	140451.8	-	486.2	-	-	-	7200
3v30n	6107.4	6106.5	0	7.1	6106.5	6106.5	0	4032.8
3v60n	11850.2	11724	2.4	21.1	17649.0	11568.5	34.5	7200
3v99n	23344.4	23076	4.7	54.8	-	22297.0	-	7200
3v180n	37503	37495.8	0.1	80.7	-	37456.0	-	7200
3v300n	61564.1	61550.8	-	145.4	-	-	-	7200
3v500n	97039.5	97013.3	-	267.2	-	-	-	7200
3v700n	140487	140463.5	-	489.1	-	-	-	7200
4v30n	6108.2	6106.5	1.7	7.1	6138.5	6005.3	2.2	7200
4v60n	11821.1	11709.5	2.4	22.1	-	11546.3	-	7200
4v99n	23299.6	22995	4.7	57.5	-	22255.0	-	7200
4v180n	37503.2	37495	-	81	-	-	-	7200
4v300n	61565.2	61554	-	147.2	-	-	-	7200
4v500n	97048.6	97016	-	263.3	-	-	-	7200
4v700n	140488.8	140485.5	-	490.2	-	-	-	7200

[a] Average objective value obtained in 20 runs
[b] Minimum (Min.) objective value obtained in 20 runs
[c] The deviation percentage of the average objective value from the LB
[d] The gap reported by CPLEX: the percentage difference between the UB and the LB with respect to the UB

When $|N| = 60$ and $|V| = 1$ and 2, CPLEX could still get an optimal solution within 2 hours whereas the EABC algorithm could obtain good feasible solutions with gaps of about 1.4% in around 20 seconds. For $|V| = 3$, CPLEX could only get a feasible solution with a gap of 34.5% and a lower bound in 2 hours. For $|V| = 4$, CPLEX could only obtain a lower bound but no feasible solution was obtained. In contrast, the EABC algorithm could get much better feasible solutions (with about 2.4% gap) using only about 20 seconds for $|V| = 3$ and 4.

When $|N|$ is greater than or equal to 99, the EABC algorithm could obtain a feasible solution within a short period of time. Even if $|N| = 700$, a feasible solution could be obtained within 500 seconds. Nevertheless, CPLEX could obtain lower and upper bounds for $|N| = 99$, 180, and 300 with $|V| = 1$ and only lower bounds for $|N| = 99$ and 180 with $|V| = 2$, 3, and 4. For other instances, CPLEX could not even obtain a lower bound in 2 hours. This shows the limitation of CPLEX and the strength of the EABC algorithm in large network applications.

In short, the EABC algorithm gives better solutions using shorter running times in all cases. Overall, this method produces high-quality solutions using short computing times.

5.3 Comparisons of total service time and maximum route duration under two objectives

This experiment was set up to compare the results of Model 1 (obtained from the EABC algorithm) with the second priority level objective of minimizing the total service time (TST) of all vehicles and those of Model 2 with the second priority level objective of minimizing the maximum route duration (MRD) of all vehicles. Table 5 gives the comparison of the results of the two models for the instances with $|V| = \{1, 2, 3\}$ under three different network sizes. For these instances, the tolerance of total demand dissatisfaction was set to 0. As expected, the average computation time increases significantly with the number of nodes and slightly with the number of vehicles used. Yet, the average computation time for solving the two models for the same instance is similar.

Table 5 Comparisons of total service time and maximum route duration under the two objectives

Instances	Minimize total service time					Minimize maximum route duration				
	Vehicles in use	Min. (mean) TST	Min. (mean) MRD	DD [a]	CPU Time [b]	Vehicles in use	Min. (mean) MRD	Min. (mean) TST	DD [a]	CPU Time [b]
1v30n	1	6106.5 (6107.4)	6106.5 (6107.4)	0	6.5	1	6106.5 (6107.4)	6106.5 (6107.4)	0	6.5
1v60n	1	11743.0 (11809.5)	11743.0 (11809.5)	0	19.8	1	11743.0 (11809.5)	11743.0 (11809.5)	0	19.1
1v99n	1	23072.5 (23228.6)	23072.5 (23228.6)	0	52.4	1	23072.5 (23228.6)	23072.5 (23228.6)	0	51.6
2v30n	1	6106.5 (6106.5)	6106.5 (6106.5)	0	6.6	2	3311.0 (3368.2)	6601.0 (6720.0)	0	7.2
2v60n	1	11684.5 (11821.2)	11684.5 (11821.2)	0	20.5	2	6136.5 (6487.3)	12232.0 (12472.8)	0	20.3
2v99n	1	23072.5 (23362.4)	23072.5 (23362.4)	0	51.2	2	12603.0 (12890.8)	25165.0 (25747.9)	0	50.1
3v30n	1	6106.5 (6107.4)	6106.5 (6107.4)	0	7.1	3	2725.0 (2844.4)	8019.0 (8384.8)	0	7.2
3v60n	1	11724.0 (11850.2)	11724.0 (11850.2)	0	22.1	3	5168.0 (5355.3)	15222.0 (15888.7)	0	20.9
3v99n	1	23076.0 (23344.4)	23076.0 (23344.4)	0	54.8	3	9722.0 (10480.5)	29087.0 (31194.5)	0	51.8

[a] Average demand dissatisfaction of 20 solutions

[b] Average CPU runtime in seconds per run

Under the objective of Model 1 that aims to minimize TST as the second priority, the results from Table 5 show that an increase in the number of available vehicles does not necessarily change the minimum TST because some of them may not be used at optimality. Besides, the best solution in every test instance only requires one vehicle in operation. From these results, it can be concluded that having more vehicles available may not be effective to reduce the minimum TST.

Under the objective of Model 2 that aims to minimize MRD as the second priority, the results from Table 5 show that the smallest MRD can be reduced with an increasing number of vehicles available (and used) but the corresponding TST increases. The results of Model 2 are also different from those obtained under the objective of Model 1 when there are at least two vehicles in use. For example, the results of instance 3v99n show that the MRD and the TST under the objective of minimizing MRD as the second priority are respectively lower and higher than those under the objective of minimizing TST as the second priority by 57.9% and 26.0%. The results demonstrate that, although the application of the objective of Model 2 can shorten the duration of the repositioning activity, it can result in a significant increase in the TST. They also illustrate the trade-off between the TST and MRD of all vehicles, meaning that it is not possible to minimize both measures simultaneously.

5.4 The effects of the upper bounds on total service time and maximum route duration

A larger TST (MRD) may be undesirable for the operator as it implies a greater fuel cost (a longer operation duration), which in turn implies a higher operating cost. However, it is not possible to minimize both TST and MRD simultaneously as shown in the last section. To limit the increase in a service time measure that is not considered the objective function, the operator can add an upper bound for that service time measure to the model concerned. In other words, an upper bound for the MRD of all vehicles is imposed in Model 1 and an upper bound for the TST of all vehicles is imposed in Model 2. However, these additions of upper bounds may affect the corresponding optimal value of TST and MRD. The objective of the experiment is, therefore, to investigate how an upper bound imposed on one measure affects the optimal value of the other measure, taking instance 2v60n as an example.

Figure 3 plots the smallest MRD against the upper bound of the TST. The results are obtained by solving Model 2 with an additional upper bound constraint on the TST. When the bound is below 11684.5 seconds (i.e., about 3.25 hours), there is no feasible solution because the upper bound constraint eliminates all solutions that satisfy other constraints in Model 2. When the bound is between 11684.5 and 11790 seconds, the optimal solution gives an MRD of 11684.5 seconds and is equivalent to the solution with the minimum TST with only one vehicle used (shown in the result of instance 2v60n that minimizes TST in Table 5). Thus, this range of upper bounds defines the minimum TST region. In the trade-off region (where the bound is between 11790 and 12232 seconds), the smallest MRD is lower when the upper bound of the TST is larger. In all optimal solutions in this region, two vehicles are used. Moreover, one of the vehicle routes is much longer than the other one because the TST is large enough to allow the operator to have a short route for one vehicle to visit a few proximate stations to reduce the service duration of the other vehicle that was originally needed to visit all stations. When the upper bound equals or exceeds 12232, the smallest MRD is 6136.5 seconds (i.e., the result of instance 2v60n that minimizes MRD) and two vehicles are in use. This result is the same as the result without the consideration of the upper bound. This is why the region is called the no-effect region. Note that the number of vehicles in use is non-decreasing with respect to the upper bound.

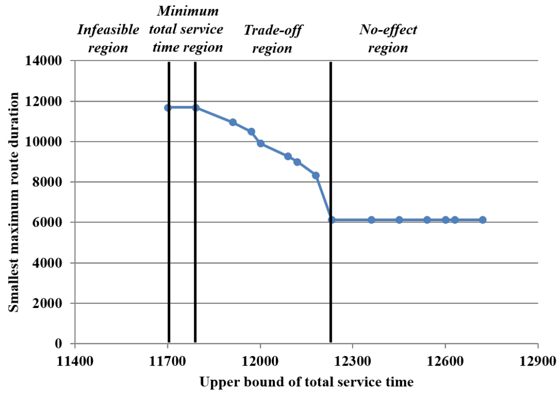


Figure 3 The smallest maximum route duration against the upper bound of the total service time

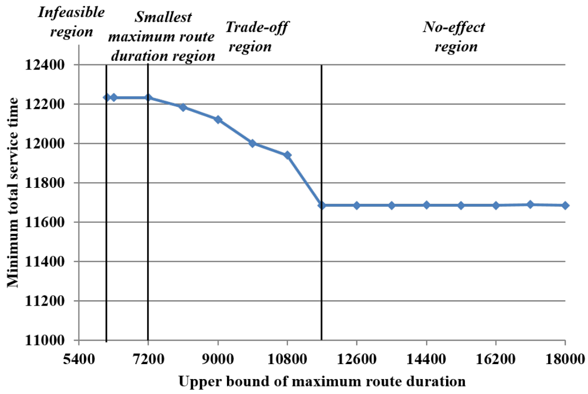
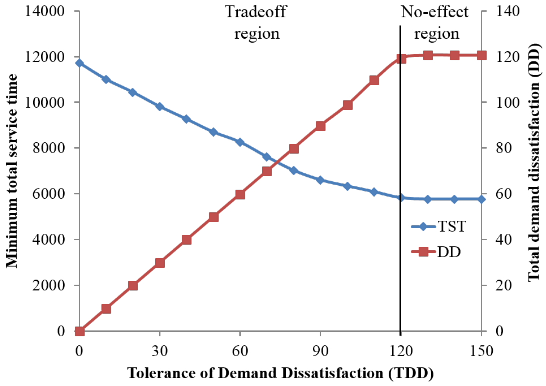


Figure 4 Minimum total service time against the upper bound of the maximum route duration

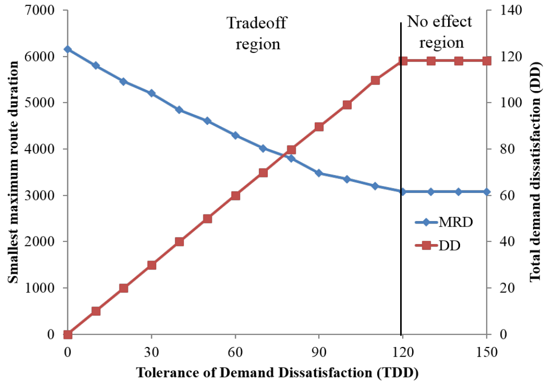
Figure 4 shows the minimum TST under different upper bounds of the MRD. The results are obtained by solving Model 1 with an additional upper bound constraint on the MRD. The shape and the trend of the line are identical to those in Figure 3. No feasible solutions were found for upper bounds below 6136.5 seconds because of the violation of the upper bound constraint by all feasible solutions of Model 1. For the upper bound of the MRD between 6136.5 to 7200 seconds, the best solution associated with each upper bound shows that both vehicles are employed for repositioning with similar travel times, which is the same as the solution with the smallest MRD with two vehicles used. Thus, the region is referred to as the smallest MRD region. In the trade-off region, it is found that one of the vehicle route durations equals to the upper bound. In the no-effect region, the TST attains its minimum (i.e., 11684.5 seconds) and only one vehicle is employed, and there is no change in the minimum TST as the upper bound increases.

5.5 The effects of the tolerance of total demand dissatisfaction

Another experiment was conducted to demonstrate the effect of the TDD on the minimum TST of all vehicles, the smallest MRD of all vehicles, and the total demand dissatisfaction of the system at optimality. Replicates of the algorithms with different TDDs were run in instance 2v60n, where the TDD value varied from 0 to 150. Figure 5(a) and Figure 5(b) show the computational results of the experiments on the smallest TST and MRD, respectively. From these two figures, it is clear that the graphs can be split into two regions: the trade-off region and the no-effect region. When $TDD = 0$, both graphs give the corresponding largest service time. The service times decrease with *TDD*, and become the smallest when the TDD reaches 120, which is the maximum total demand dissatisfaction in the system. After that, a further increase in *TDD* does not have any effect on the two smallest service time values. For the total demand dissatisfaction (denoted as DD in the two graphs), both graphs show that DD increases linearly with the value of *TDD* in the trade-off region, and remains unchanged in the no-effect region.



(a) Minimum total service time and total demand dissatisfaction against the TDD



(b) Smallest maximum route duration and total demand dissatisfaction against the TDD

Figure 5 TST, MRD, and total demand dissatisfaction against the tolerance of total demand dissatisfaction

To clearly show the trade-off between the total demand dissatisfaction and the minimum TST, Figure 6, derived from Figure 5(a), is given. In this study, the total demand dissatisfaction is expressed as $\max\left[\sum_{j\in N}\left(d_j-\sum_{v\in V} r_{jv}\right)-TDD,0\right]$. When the first term in the square bracket is positive, a large value is added to the objective value due to the coefficient, M. When TDD gets larger, the value of $\sum_{j\in N}\left(d_j-\sum_{v\in V} r_{jv}\right)$ can also be larger to ensure that the first term in the square bracket is positive. This implies that fewer bikes are required to be loaded, and thus the minimum TST can be lower.

Figure 7 extracts the total demand dissatisfaction trend in Figure 5(a) and displays together with the total demand satisfaction under different TDD values. The two figures show that an increase in TDD does not only reduce the minimum TST but also the total demand satisfaction of the system.

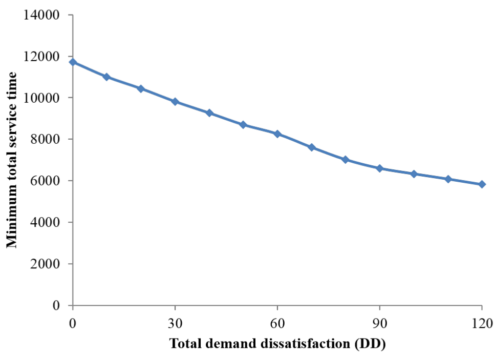


Figure 6 Trade-off between total demand dissatisfaction against minimum total service time

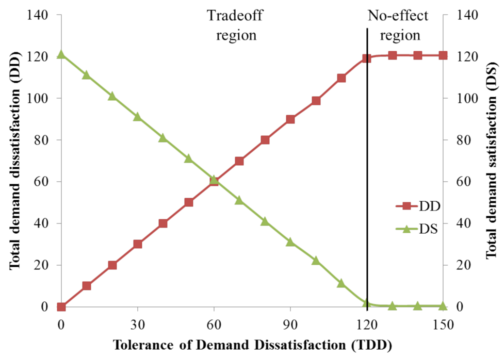


Figure 7 Total demand satisfaction and dissatisfaction against the tolerance of total demand dissatisfaction

5.6 The effects of the fleet size

This section discusses the effect of the fleet size of the repositioning vehicles on the two service times at optimality under various values of the tolerance of total demand dissatisfaction. For simplicity, a small number of bike stations (n = 30) is used and three levels of TDD (i.e., 0, 9, and 29) are used. The fleet size varies from 2 to 5 vehicles.

Table 6 shows the results of using 2 to 5 vehicles to reposition the bikes under the three levels of *TDD*. The CPU times for all instances are similar and apparently independent of the number of vehicles used and the *TDD* level. These results also verify the phenomenon in Section 5.3 that for a given TDD, an increase in the number of vehicles provided may not change the minimum TST but can decrease the smallest MRD. It is noted that the smallest MRD cannot drop to zero as the fleet size increases because the travel times from the depot to all stations cannot be zero.

Table 6 The effects of the fleet size

		Minimize total service time					Minimize maximum route duration				
Fleet Size	TDD	Veh[a]	DD[b]	Min. (mean) TST	Min. (mean) MRD	CPU Time[c]	Veh[a]	DD[b]	Min. (mean) MRD	Min. (mean) TST	CPU Time[c]
2	29	2	29	6597.5 (6597.5)	4303.0 (4303.0)	7.8	2	29	3387.5 (3394.0)	6750.5 (6769.4)	7.8
2	9	2	29	Infeasible[d]			2	29	Infeasible[d]		
2	0	2	29	Infeasible[d]			2	29	Infeasible[d]		
3	29	2	29	6597.5 (6597.5)	4303.0 (4303.0)	7.6	3	29	2671.5 (2699.3)	7945.7 (7983.5)	7.6
3	9	3	9	8484.0 (8492.5)	3943.5 (4154.8)	7.6	3	9	3101.0 (3111.1)	9242.0 (9300.8)	7.6
3	0	3	9	Infeasible[d]			3	9	Infeasible[d]		
4	29	2	29	6597.5 (6597.5)	4303.0 (4303.0)	7.7	4	29	2620.5 (2629.6)	10158.5 (10375.6)	7.6
4	9	3	9	8484.0 (8504.6)	3943.5 (4303.6)	7.7	4	9	2776.0 (2811.8)	11033.0 (11152.6)	7.6
4	0	4	0	9846.5 (9901.3)	3811.5 (4233.2)	7.8	4	0	2939.5 (3003.0)	11597.5 (11821.8)	7.7
5	29	2	29	6597.5 (6597.5)	4303.0 (4303.0)	7.6	5	29	2620.5 (2620.5)	12433.5 (12760.5)	7.5
5	9	3	9	8484.0 (8517.8)	3943.5 (4287.1)	7.6	5	9	2694.5 (2744.8)	13397.5 (13409.7)	7.5
5	0	4	0	9846.5 (9897.2)	3811.5 (4320.9)	7.6	5	0	2866.5 (2883.0)	14153.9 (14173.9)	7.5

[a] Number of vehicles used for the repositioning activity
[b] The mean total demand dissatisfaction for 20 runs
[c] Average CPU runtime in seconds per run
[d] Infeasible is defined as "no feasible solutions have their total demand dissatisfactions less than their tolerance (i.e., no feasible solutions with DD less than *TDD*)"

Table 6 illustrates the trade-off between the number of operating vehicles and the tolerance of total demand dissatisfaction. The smaller the TDD is, the more the number of operating vehicles is required to have feasible solutions. It is in accordance with the intuition since for the instances with a stricter TDD, the overall required (vehicle) capacity for bike repositioning is larger. Instances with 2 or 3 vehicles at *TDD* = 0 are examples of such a phenomenon.

Table 6 demonstrates that using more operating vehicles can reduce the smallest MRD of the whole operation for a given TDD under the objective of minimizing MRD as the second priority. For example, when comparing with the 4-vehicle instance for *TDD* = 0, the 5-vehicle instance gives the smallest MRD 73 seconds smaller (about 1.2 minutes). However, once the minimum MRD is reached, a further increase in the number of vehicles provided only results in an increase in the TST of the repositioning activities. For instance, for the objective of minimizing MRD as the second priority with *TDD* = 29, the smallest MRDs of the 4- and 5-vehicle instances are the same, but the TST for the 5-vehicle instance is 19.8% more than the 4-vehicle one.

Perfect balance is more costly compared with slight imbalance as the minimum TST for perfect balance is longer. From the optimal solutions that minimize TST as the second priority, it is found that the minimum total demand dissatisfaction equals 9 and 0 for all 3- and 4-vehicle instances considered, respectively. Meanwhile, the minimum TST found in the corresponding 3-vehicle case (8484.0 seconds) is comparably lower than that in the 4-vehicle case (9846.5 seconds). This is equivalent to the average time reduction of 151.4 seconds (2.5 minutes) for each unsatisfied demand.

The above observations leave the operator a critical problem about determining the optimal number of vehicles to hire. If the operator aims at minimizing TST as the second priority, the number of vehicles used should be equal to the minimum number of vehicles that can provide feasible solutions for a particular level of *TDD*. A similar approach is adopted if the operator aims at on minimizing MRD as the second priority. However, if both TST (or operation cost in general) and total demand dissatisfaction are both considered in the first level optimization, the determination requires the discretion of the operator to set a weight for TST and a weight for total demand dissatisfaction because there is a trade-off between total demand dissatisfaction and TST.

6 CONCLUSION

This study examines a static multiple-vehicle bike repositioning problem in a bike-sharing system, which is to determine vehicle routes and loading and unloading quantities at each visited node to minimize first the positive deviation from the tolerance of the total demand dissatisfaction of the system and then the total (or maximum) service time of the vehicles. To solve the problem, this study uses the enhanced artificial bee colony (EABC) algorithm that incorporates a new set of optimal loading and unloading strategies to solve the sub-problem efficiently. The results demonstrate that the EABC algorithm outperforms CPLEX in terms of both solution quality and computational speed.

The results also demonstrate the following.

- An increase in the number of available vehicles does not necessarily change the minimum total service time because some of them may not be used at optimality.
- The smallest maximum route duration can be reduced by increasing the number of vehicles used.
- There is a trade-off between the total service time and maximum route duration of all vehicles.
- Introducing an upper bound on one service time measure can lead to no feasible solution or have no effect on another service time measure in some cases but in other cases, a looser upper bound can lead to a smaller minimum service time.
- The two smallest service times are non-increasing with respect to the tolerance of total demand dissatisfaction (TDD). They are decreasing when the TDD is less than the maximum total demand dissatisfaction in the system.
- The total demand dissatisfaction is non-decreasing with respect to the TDD. It is increasing when the TDD is less than the maximum total demand dissatisfaction in the system.
- There is a trade-off between the total demand dissatisfaction and the total service time (operation cost), implying that perfect balance is more costly than slight imbalance.
- There is a trade-off between the number of operating vehicles provided and the TDD. The smaller the TDD is, the larger the number of operating vehicles is required to have feasible solutions.

Last but not least, the proposed methodology herein relies on an accurate estimation of the target bike and space inventory levels as inputs. The real data from bike-sharing systems can only provide realized demand, not the true demand. How to estimate the true demand and hence the space inventory level accurately is an important research question. O'Mahony and Shmoys (2015) provided a method to determine the lower bound of the true demand for this purpose and assumed that the lower bound is a good proxy for the true demand. It is expected that other accurate estimation methods will also be proposed in the future.

ACKNOWLEDGEMENTS

This research was jointly supported by a grant from the National Natural Science Foundation of China (No. 71771194) and a grant (No. 201511159095) from the University Research Committee of the University of Hong Kong. The authors are grateful to the two reviewers for their constructive comments.

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*, *62*, 163-175.

Angeloudis, P., Hu, J., and Bell, M. G. H., 2014. A strategic repositioning algorithm for bicycle-sharing schemes. *Transportmetrica A: Transport Science*, *10*(8), pp.759-774.

Benchimol, M., Benchimol, P., Chappert, B., De La Taille, A., Laroche, F., Meunier, F., and Robinet, L., 2011. Balancing the stations of a self service "bike hire" system. *RAIRO-Operations Research*, *45*(1), pp.37-61.

Brinkmann, J., Ulmer, M. W., and Mattfeld, D. C., 2015. Short-term strategies for stochastic inventory routing in bike sharing systems. *Transportation Research Procedia*, *10*, pp.364–373.

Caggiani, L., and Ottomanelli, M., 2012. A modular soft computing based method for vehicles repositioning in bike-sharing systems. *Procedia-Social and Behavioral Sciences*, *54*, pp.675-684.

Chemla, D., Meunier, F., and Calvo, R. W., 2013. Bike sharing systems: Solving the static rebalancing problem. *Discrete Optimization*, *10*(2), pp.120-146.

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

Cruz, F., Subramanian, A., Bruck, B. P., and Iori, M., 2017. A heuristic algorithm for a single vehicle static bike sharing rebalancing problem. *Computers and Operations Research*, 79, pp.19-33.

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

Dell'Amico, M., Iori, M., Novellani, S., and Stützle, T., 2016. A destroy and repair algorithm for the bike sharing rebalancing problem. *Computers and Operations Research*, *71*, pp.149-162.

Di Gaspero, L., Rendl, A., and Urli, T., 2013a. A hybrid ACO+ CP for balancing bicycle sharing systems. In *Hybrid Metaheuristics*, *Lecture Notes in Computer Science*, Volume 7919 (pp.198-212). Springer Berlin Heidelberg.

Di Gaspero, L., Rendl, A., and Urli, T., 2013b. Constraint-based approaches for balancing bike sharing systems. In *Principles and Practice of Constraint Programming, Lecture Notes in Computer Science*, Volume 8124 (pp.758-773). Springer Berlin Heidelberg.

Di Gaspero, L., Rendl, A., and Urli, T., 2016. Balancing bike sharing systems with constraint programming. *Constraints*, *21*(2), pp.318-348.

Erdoğan, G., Laporte, G., and Calvo, R. W., 2014. The static bicycle relocation problem with demand intervals. *European Journal of Operational Research*, *238*(2), pp.451-457.

Erdoğan, G., Battarra, M., and Calvo, R. W., 2015. An exact algorithm for the static rebalancing problem arising in bicycle sharing systems. *European Journal of Operational Research*, 245(3), pp.667-679.

Forma, I. A., Raviv, T., and Tzur, M., 2015. A 3-step math heuristic for the static repositioning problem in bike-sharing systems. *Transportation Research Part B: Methodological*, *71*, pp.230-247.

Ho, S. C., and Szeto, W. Y., 2014. Solving a static repositioning problem in bike-sharing systems using iterated tabu search. *Transportation Research Part E: Logistics and Transportation Review*, *69*, pp.180-198.

Ho, S. C., and Szeto, W. Y., 2017. A hybrid large neighborhood search for the static multi-vehicle bike-repositioning problem. *Transportation Research Part B: Methodological*, *95*, pp.340-363.

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 and Industrial Engineering*, *95*, pp.41-52.

Karaboga, D., 2005. An idea based on honey bee swarm for numerical optimization. Technical Report TR06, Erciyes University; 2005.

Karaboga, D., and Basturk, B., 2007. A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *Journal of Global Optimization*, *39*(3), pp.459-471.

Karaboga, D., and Basturk, B., 2008. On the performance of artificial bee colony (ABC) algorithm. *Applied Soft Computing*, *8*(1), pp.687-697.

Karaboga, D., Gorkemli, B., Ozturk, C., and Karaboga, N. 2014. A comprehensive survey: artificial bee colony (ABC) algorithm and applications. *Artificial Intelligence Review*, *42*(1), pp.21-57.

Kloimüllner, C., Papazek, P., Hu, B., and Raidl, G. R., 2014. Balancing bicycle sharing systems: An approach for the dynamic case. In *Evolutionary Computation in Combinatorial Optimisation*, *Lecture Notes in Computer Science*, Volume 8600 (pp.73-84), Springer Berlin Heidelberg.

Li, Y., Szeto, W. Y., Long, J. C., and Shui, C. S., 2016. A multiple type bike repositioning problem. *Transportation Research Part B*, *90*, pp.263-278.

Lin, J. H., and Chou, T.C., 2012. A geo-aware and VRP-based public bicycle redistribution system. *International Journal of Vehicular Technology*, *2012*: 14 pages

Meddin, R., and DeMaio, P., 2017. The bike-sharing world map. Retrieved from http://www.bikesharingworld.com/ [access on 2 November 2017]

Nair, R., Miller-Hooks, E., Hampshire, R. C., and Bušić, A., 2013. Large-scale vehicle sharing systems: analysis of Vélib'. *International Journal of Sustainable Transportation*, *7*(1), pp.85-106.

Nair, R., and Miller-Hooks, E., 2011. Fleet management for vehicle sharing operations. *Transportation Science*, *45*(4), pp.524-540.

O'Mahony, E., and Shmoys, D. B. 2015. Data analysis and optimization for (Citi) bike sharing. In *Proceedings of the 29th AAAI Conference of Artificial Intelligence* (pp. 687-694).

Raidl, G. R., Hu, B., Rainer-Harbach, M., and Papazek, P., 2013. Balancing bicycle sharing systems: Improving a VNS by efficiently determining optimal loading operations. In *Hybrid Metaheuristics*, *Lecture Notes in Computer Science,* Volume 7919 (pp.130-143). Springer Berlin Heidelberg.

Rainer-Harbach, M., Papazek, P., Hu, B., and Raidl, G. R., 2013. Balancing bicycle sharing systems: A variable neighborhood search approach. In *Evolutionary Computation in Combinatorial Optimization*, *Lecture Notes in Computer Science*, Volume 7832 (pp.121-132). Springer Berlin Heidelberg.

Rainer-Harbach, M., Papazek, P., Raidl, G. R., Hu, B., and Kloimüllner, C., 2015. PILOT, GRASP, and VNS approaches for the static balancing of bicycle sharing systems. *Journal of Global*

Optimization, *63*(3), pp.597-629.

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*, *2*(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*, *72*, pp.192-209.

Schuijbroek, J., Hampshire, R., and van Hoeve, W. J., 2017. Inventory rebalancing and vehicle routing in bike sharing systems, *European Journal of Operational Research*, *257*(3), pp.992–1004.

Shi, X., Zhao, F., and Gong, Y., 2009. Genetic algorithm for the one-commodity pickup-and-delivery vehicle routing problem. In *2009 IEEE International Conference on Intelligent Computing and Intelligent Systems*, Volume 1 (pp.175-179), IEEE.

Shui, C.S., and Szeto, W.Y., 2017. Dynamic green bike repositioning problem–A hybrid rolling horizon artificial bee colony algorithm approach. *Transportation Research Part D: Transport and Environment*, in press.

Szeto, W. Y. and Jiang, Y., 2014. Transit route and frequency design: Bi-level modeling and hybrid artificial bee colony algorithm approach. *Transportation Research Part B*, *67*, pp.235-263.

Szeto, W. Y., Wu, Y., and Ho, S. C., 2011. An artificial bee colony algorithm for the capacitated vehicle routing problem. *European Journal of Operational Research*, *215*(1), pp.126-135.

Szeto, W. Y., Liu, Y., and Ho, S. C., 2016. Chemical reaction optimization for solving a static bike repositioning problem. *Transportation Research Part D*, *47*, pp.104-135.

Ting, C. K., and Liao, X. L., 2013. The selective pickup and delivery problem: formulation and a memetic algorithm. *International Journal of Production Economics*, *141*(1), pp.199-211.

Zhang, D., Yu, C., Desai, J., Lau, H. Y. K., and Srivathsan, S., 2016. A time-space network flow approach to dynamic repositioning in bicycle sharing systems. *Transportation Research Part B: Methodological*, *103*, pp.188-207.

APPENDIX A: PROOFS OF PROPOSITIONS

Appendix A.1: Proof of Proposition 1

To minimize the total demand dissatisfaction along the route, the demand dissatisfaction at each drop-off node in the route must be minimized. To minimize the demand dissatisfaction at each drop-off (bike deficiency) node, the number of bikes on the vehicle heading to each bike deficiency node must always be maximized so as to minimize the bike deficiency of that node and the subsequent bike deficiency nodes, which implies to pick up bikes at each pickup node as many as possible. Due to constraints (4), (5), (7), and (9), the maximum number of bikes loaded onto the vehicle at a station is constrained by either case: all excess bikes at the station (i.e., s_j) are picked up or the remaining vehicle capacity is reached (i.e., $Q - \sum_{i \in N_o \setminus \{j\}} q_{ijv}$). The first case implies $p_{jv} \le s_j$ (i.e., constraint (5)). The expression for the second case, i.e., remaining vehicle capacity, is deduced from constraints (4), (7), and (9) as follows. Combining constraints (7) and (9), we have $\sum_{v \in V} \sum_{i \in N_0 \setminus \{j\}} q_{ijv} \le Q$. Moreover, in any pickup node, which does not have drop-off activities (i.e., $r_{jv} = 0, \forall v \in V$), constraint (4) can be rewritten as $p_{jv} = \sum_{i \in N_0 \setminus \{j\}} q_{jiv} - \sum_{i \in N_0 \setminus \{j\}} q_{ijv}$. As the first term on the right side is not greater than Q, the right side is not greater than $Q-\sum_{i\in N_o\backslash\{j\}} q_{ijv}$, leading to $p_{jv}\leq Q-\sum_{i\in N_o\backslash\{j\}} q_{ijv}$. The maximum number of bikes loaded at each pickup station is, therefore, the smaller one between the two terms, s_j and $Q-\sum_{i\in N_0\backslash\{j\}} q_{ijv}$, which is equation (17a).

At a bike deficiency station, the number of bikes to be unloaded from the vehicle must always be maximized to minimize the bike deficiency, which implies to drop bikes off as many as possible. Due to constraints (4), (6), and (11), the maximum number of bikes unloaded is constrained by two situations: either the bike deficit at that station (i.e., d_j) is cleared, or all bikes on the vehicle are unloaded (i.e., $\sum_{i\in N_0\backslash\{j\}} q_{ijv}$). The first situation implies $r_{jv}\leq d_j$ (i.e., inequality (6)). The second situation is related to the number of bikes on the vehicle that is deduced from constraints (4) and (11) as follows. At any bike deficiency station, which does not have pickup activities (i.e., $p_{jv}=0,\forall v\in V$), constraint (4) can be rewritten as $r_{jv}=\sum_{i\in N_0\backslash\{j\}} q_{ijv}-\sum_{i\in N_0\backslash\{j\}} q_{jiv}$. Constraint (11) requires that $q_{ijv}\geq 0$, so $r_{jv}=\sum_{i\in N_0\backslash\{j\}} q_{ijv}-\sum_{i\in N_0\backslash\{j\}} q_{jiv}\leq \sum_{i\in N_0\backslash\{j\}} q_{ijv}$. The maximum number of bikes unloaded at each drop-off station is obtained by finding the expression that satisfies the conditions $r_{jv}\leq \sum_{i\in N_0\backslash\{j\}} q_{ijv}$ and $r_j\leq d_j$ simultaneously, which gives equation (18a). This completes the proof.■

Appendix A.2: Proof of Proposition 2

In the sub-problem, all vehicle routes are known. Therefore, the values of all x_{ijv} are known and can be substituted into constraint (7). Consequently, the right-hand side of constraint (7) can be non-zero only if nodes *i* and *j* are in the same route and adjacent to each other. If we define i_ε^v as the *ε*-th visited node in route *v* with $|J_v|$ stations, where $\varepsilon\in\{1,\ldots,|J_v|\}$, then constraint (7) can be rewritten as

$$q_{i_\varepsilon^v,i_{\varepsilon+1}^v,v}\leq Q, \qquad \forall i_\varepsilon^v,i_{\varepsilon+1}^v\in J_v, \text{ and} \tag{7a}$$

$$q_{0,i_1^v,v}\leq Q. \tag{7b}$$

Using the new set of notations associated with artificial nodes, we consider four mutually exclusive but collective exhaustive cases for constraint (7a): (i) $i_\varepsilon^v\in R_{a-1}^v$, $i_{\varepsilon+1}^v\in P_a^v\cap\{0\}$; (ii) $i_\varepsilon^v\in P_a^v$, $i_{\varepsilon+1}^v\in P_a^v$; (iii) $i_\varepsilon^v\in P_a^v$, $i_{\varepsilon+1}^v\in R_a^v$; (iv) $i_\varepsilon^v\in R_a^v$, $i_{\varepsilon+1}^v\in R_a^v$.

For case (i), constraint (7a) can be written as

$$q_{a-1,a}^v\leq Q, \qquad \forall a\in\{2,\ldots,n^v+1\}. \tag{31a}$$

Based on equation (28), the above constraint can be re-expressed as

$$q_{a-1,a}^v+p_a^v-r_a^v\leq Q, \qquad \forall a\in\{1,\ldots,n^v\}. \tag{31b}$$

For case (ii), constraint (7a) can be respectively written as

$$q_{a-1,a}^v+\sum_{f'=1}^{f-1}p_{a,f'}^v\leq Q, \qquad \forall a\in\{1,\ldots,n^v\}, f\in\{2,\ldots,|P_a^v|\}. \tag{31c}$$

where the left hand side of condition (31c) is the bike load on vehicle *v* traveling between the (*f*-1)-th and the *f*-th node of the *a*-th artificial pickup node. For cases (iii) and (iv), constraint (7a) can be respectively written as

$$q^v_{a-1,a} + p^v_a \le Q, \qquad \forall a \in \{1,\dots,n^v\}, \text{ and} \tag{31d}$$

$$q^v_{a-1,a} + \left(p^v_a - \sum_{f'=1}^{f-1} r^v_{a,f'}\right) \le Q, \qquad \forall a \in \{1,\dots,n^v\}, f \in \{2,\dots,|R^v_a|\}. \tag{31e}$$

Using the new set of notations associated with artificial nodes, constraint (7b) becomes $q^v_{0,1} \le Q$, which is a special case of

$$q^v_{a-1,a} \le Q, \qquad a = 1. \tag{31f}$$

By comparing conditions (31b)-(31f), it is observed that the main difference between them is the second term on their left-hand sides (i.e., $p^v_a - r^v_a$, $\sum_{f'=1}^{f-1} p^v_{a,f'}$, p^v_a, $p^v_a - \sum_{f'=1}^{f-1} r^v_{a,f'}$, and 0). By definition, we have $0 \le \sum_{f'=1}^{f-1} p^v_{a,f'} \le p^v_a$ and $0 \le \sum_{f'=1}^{f-1} r^v_{a,f'} \le r^v_a$, $\forall f \in \{2,\dots,|P^v_a|\}$, $\forall a \in \{1,\dots,n^v\}$. Therefore, p^v_a is not less than the other four terms on the left-hand sides (i.e., $p^v_a \ge \sum_{f'=1}^{f-1} p^v_{a,f'}$, $p^v_a \ge p^v_a - \sum_{f'=1}^{f-1} r^v_{a,f'}$, $p^v_a \ge p^v_a - r^v_a$, and $p^v_a \ge 0$), which implies that when condition (31d) holds, conditions (31b), (31c), (31e), and (31f) must hold. Hence, condition (31d) (i.e., condition (31)) becomes the condition that corresponds to constraint (7). This completes the proof. ■

Appendix A.3: Proof of Proposition 3

In (41), when $p^{v,2}_a$ equals $Q - \Delta^{v,2}_{a-1}$ or $d^v_a + \phi^v_{a+1} - \Delta^{v,2}_a$, it implies $\min\left(Q - \Delta^{v,2}_{a-1}, d^v_a + \phi^v_{a+1} - \Delta^{v,2}_{a-1}\right) \le s^v_a$, and thus $p^{v,2}_a \le s^v_a$, which satisfies (29); when $p^{v,2}_a = s^v_a$, it also satisfies (29) as $p^{v,2}_a = s^v_a$ is the special case of $p^{v,2}_a \le s^v_a$.

In (42), when $r^{v,2}_a$ equals $p^{v,2}_a + \Delta^{v,2}_{a-1}$, it implies $p^{v,2}_a + \Delta^{v,2}_{a-1} \le d^v_a$, and thus $r^{v,2}_a \le d^v_a$, which satisfies (30); when $r^{v,2}_a = d^v_a$, it also satisfies (30) because $r^{v,2}_a = d^v_a$ is the special case of $r^{v,2}_a \le d^v_a$.

To show that (41) and (42) satisfy (37), we firstly consider, for all $a \in \{1,\dots,n^v\}$,

$$\begin{aligned} & p^{v,2}_a - r^{v,2}_a \\ &= p^{v,2}_a - \min\left(d^v_a, p^{v,2}_a + \Delta^{v,2}_{a-1}\right) (\because (42)) \\ &= \max\left(p^{v,2}_a - d^v_a, -\Delta^{v,2}_{a-1}\right) \\ &= \max\left(\min\left(s^v_a, Q - \Delta^{v,2}_{a-1}, d^v_a + \phi^v_{a+1} - \Delta^{v,2}_{a-1}\right) - d^v_a, -\Delta^{v,2}_{a-1}\right). \end{aligned}$$

When $a = 1$, given that $\Delta^{v,2}_0 = 0$, $p^{v,2}_1 - r^{v,2}_1 = \Delta^{v,2}_1 = \max\left(\min\left(s^v_1 - d^v_1, Q - d^v_1, \phi^v_2\right), 0\right) \ge 0$.

When $a > 1$,

$$\sum_{b=1}^{a}\left(p_b^{v,2}-r_b^{v,2}\right)=\left(p_a^{v,2}-r_a^{v,2}\right)+\Delta_{a-1}^{v,2}$$
$$=\max\left(\min\left(s_a^v,Q-\Delta_{a-1}^{v,2},d_a^v+\phi_{a+1}^v-\Delta_{a-1}^{v,2}\right)-d_a^v,-\Delta_{a-1}^{v,2}\right)+\Delta_{a-1}^{v,2}$$
$$=\max\left(\min\left(s_a^v+\Delta_{a-1}^{v,2}-d_a^v,Q-d_a^v,\phi_{a+1}^v\right),0\right)\geq 0.$$

Therefore, we have

$$\Delta_a^{v,2}=\sum_{b=1}^{a}\left(p_b^{v,2}-r_b^{v,2}\right)\geq 0, \text{ for all } a\in\{1,\ldots,n^v\}, \text{ and} \tag{37a}$$

$$\Delta_a^{v,2}=\max\left(\min\left(s_a^v+\Delta_{a-1}^{v,2}-d_a^v,Q-d_a^v,\phi_{a+1}^v\right),0\right), \text{ for all } a\in\{1,\ldots,n^v\}, \tag{37b}$$

and hence (41) and (42) satisfy (37).

Substituting (41) and (42) into the left-hand side of constraint (36), we have

$$Q-\left(p_a^{v,2}+\Delta_{a-1}^{v,2}\right)=Q-\left[\min\left(s_a^v,Q-\Delta_{a-1}^{v,2},d_a^v+\phi_{a+1}^v-\Delta_{a-1}^{v,2}\right)+\Delta_{a-1}^{v,2}\right]$$
$$=Q-\left[\min\left(s_a^v+\Delta_{a-1}^{v,2},Q,d_a^v+\phi_{a+1}^v\right)\right]$$

Because of (37), $\Delta_a^{v,2}\geq 0$. Moreover, Q, s_a^v, d_a^v, and ϕ_{a+1}^v are non-negative. Therefore, $\left[\min\left(s_a^v+\Delta_{a-1}^{v,2},Q,d_a^v+\phi_{a+1}^v\right)\right]$ is non-negative. By definition, $\left[\min\left(s_a^v+\Delta_{a-1}^{v,2},Q,d_a^v+\phi_{a+1}^v\right)\right]$ cannot be larger than Q. Therefore, $Q-\left[\min\left(s_a^v+\Delta_{a-1}^{v,2},Q,d_a^v+\phi_{a+1}^v\right)\right]\geq 0$, meaning that (41) and (42) satisfy (36).

To show that (41) satisfies (38), i.e., $p_a^{v,2}=\min\left(s_a^v,Q-\Delta_{a-1}^{v,2},d_a^v+\phi_{a+1}^v-\Delta_{a-1}^{v,2}\right)\geq 0$, we do four steps. We firstly obtain $\Delta_{a-1}^{v,2}$. From (37b), $\Delta_a^{v,2}=\max\left(\min\left(s_a^v+\Delta_{a-1}^{v,2}-d_a^v,Q-d_a^v,\phi_{a+1}^v\right),0\right)$. Replacing a with a-1, we have $\Delta_{a-1}^{v,2}=\max\left(\min\left(s_{a-1}^v+\Delta_{a-2}^{v,2}-d_{a-1}^v,Q-d_{a-1}^v,\phi_a^v\right),0\right)$.

Secondly, we re-express $p_a^{v,2}$ as $\min\left(s_a^v,\min\left(Q,d_a^v+\phi_{a+1}^v\right)-\Delta_{a-1}^{v,2}\right)$ and replace $\Delta_{a-1}^{v,2}$ with $\max\left(\min\left(s_{a-1}^v+\Delta_{a-2}^{v,2}-d_{a-1}^v,Q-d_{a-1}^v,\phi_a^v\right),0\right)$, giving

$$p_a^{v,2}$$
$$=\min\left(s_a^v,\min\left(Q,d_a^v+\phi_{a+1}^v\right)-\max\left(\min\left(s_{a-1}^v+\Delta_{a-2}^{v,2}-d_{a-1}^v,Q-d_{a-1}^v,\phi_a^v\right),0\right)\right)$$
$$=\min\left(s_a^v,\min\left(Q,d_a^v+\phi_{a+1}^v\right)+\min\left(\max\left(d_{a-1}^v-s_{a-1}^v-\Delta_{a-2}^{v,2},d_{a-1}^v-Q,-\phi_a^v\right),0\right)\right)$$
$$=\min\left(s_a^v,Q,d_a^v+\phi_{a+1}^v,Q+\max\left(d_{a-1}^v-s_{a-1}^v-\Delta_{a-2}^{v,2},d_{a-1}^v-Q,-\phi_a^v\right),d_a^v+\phi_{a+1}^v+\max\left(d_{a-1}^v-s_{a-1}^v-\Delta_{a-2}^{v,2},d_{a-1}^v-Q,-\phi_a^v\right)\right).$$

Hence,

$$p_a^{v,2}=\min\begin{pmatrix} s_a^v, \\ Q, \\ d_a^v+\phi_{a+1}^v, \\ Q+\max\left(d_{a-1}^v-s_{a-1}^v-\Delta_{a-2}^{v,2},d_{a-1}^v-Q,-\phi_a^v\right), \\ d_a^v+\phi_{a+1}^v+\max\left(d_{a-1}^v-s_{a-1}^v-\Delta_{a-2}^{v,2},d_{a-1}^v-Q,-\phi_a^v\right)\end{pmatrix}. \tag{41a}$$

Thirdly, we aim to show that all terms in (41a) are non-negative. The first and second terms are obviously non-negative. Based on (44), $\phi^v_{a+1} \geq 0$; d^v_a is non-negative. Hence, the third term is non-negative. For the fourth term in the minimization bracket in (50a), we can do simplification as

$$Q + \max\left(d^v_{a-1} - s^v_{a-1} - \Delta^{v,2}_{a-2}, d^v_{a-1} - Q, -\phi^v_a\right)$$
$$= \max\left(Q + d^v_{a-1} - s^v_{a-1} - \Delta^{v,2}_{a-2}, d^v_{a-1}, Q - \phi^v_a\right).$$

In this maximization term, when $d^v_{a-1} > \max\left(Q + d^v_{a-1} - s^v_{a-1} - \Delta^{v,2}_{a-2}, Q - \phi^v_a\right)$, the fourth term in the minimization bracket in (41a), which equals d^v_{a-1}, must be non-negative as d^v_{a-1} is non-negative. Besides, when $d^v_{a-1} \leq \max\left(Q + d^v_{a-1} - s^v_{a-1} - \Delta^{v,2}_{a-2}, Q - \phi^v_a\right)$, the fourth term in the minimization bracket in (41a), which equals $\max\left(Q + d^v_{a-1} - s^v_{a-1} - \Delta^{v,2}_{a-2}, Q - \phi^v_a\right)$, must be non-negative as $0 \leq d^v_{a-1} \leq \max\left(Q + d^v_{a-1} - s^v_{a-1} - \Delta^{v,2}_{a-2}, Q - \phi^v_a\right)$. This implies that the fourth term in (41a) must be non-negative. Similarly, the fifth term in the minimization bracket in (41a) can also undergo simplification:

$$d^v_a + \phi^v_{a+1} + \max\left(d^v_{a-1} - s^v_{a-1} - \Delta^{v,2}_{a-2}, d^v_{a-1} - Q, -\phi^v_a\right)$$
$$= \max\left(d^v_a + \phi^v_{a+1} + d^v_{a-1} - s^v_{a-1} - \Delta^{v,2}_{a-2}, d^v_a + \phi^v_{a+1} + d^v_{a-1} - Q, d^v_a + \phi^v_{a+1} - \phi^v_a\right). \tag{41b}$$

Consider the third term in the maximization bracket on the right-hand side of (41b), it can be expressed as $d^v_a + \phi^v_{a+1} - \left[\phi^v_{a+1} + \max\left(d^v_a - s^v_a, d^v_a - Q, 0\right)\right]$ because $\phi^v_a = \sum_{b=a}^{n^v} \max\left(d^v_b - s^v_b, d^v_b - Q, 0\right)$ (from (44)), which can be rewritten as $\phi^v_a = \phi^v_{a+1} + \max\left(d^v_a - s^v_a, d^v_a - Q, 0\right)$. Hence, we have

$$d^v_a + \phi^v_{a+1} - \phi^v_a = d^v_a + \phi^v_{a+1} - \left[\phi^v_{a+1} + \max\left(d^v_a - s^v_a, d^v_a - Q, 0\right)\right]$$
$$= d^v_a + \max\left(s^v_a - d^v_a, Q - d^v_a, 0\right)$$
$$= \max\left(s^v_a, Q, d^v_a\right) \geq 0. \tag{41c}$$

When the third term in the maximization bracket, $d^v_a + \phi^v_{a+1} - \phi^v_a$, on the right-hand side of (41b) is the maximum among all three terms in the maximization bracket of (41b), the fifth term in (41a) becomes $d^v_a + \phi^v_{a+1} - \phi^v_a$ and must be non-negative according to (41c). Moreover, when the maximum of the first and second terms of (41b) is greater than or equal to the third term in (41b), i.e., $\max\left(d^v_a + \phi^v_{a+1} + d^v_{a-1} - s^v_{a-1} - \Delta^{v,2}_{a-2}, d^v_a + \phi^v_{a+1} + d^v_{a-1} - Q\right) \geq d^v_a + \phi^v_{a+1} - \phi^v_a$, the fifth term in (41a) becomes $\max\left(d^v_a + \phi^v_{a+1} + d^v_{a-1} - s^v_{a-1} - \Delta^{v,2}_{a-2}, d^v_a + \phi^v_{a+1} + d^v_{a-1} - Q\right)$ and must also be non-negative as the fifth term in (41a) is at least equal to $d^v_a + \phi^v_{a+1} - \phi^v_a$ which is non-negative according to (41c).

Fourthly, s^v_a, Q, and d^v_a are integers for all $a \in \{1,\ldots,n^v\}$. For ϕ^v_{a+1}, (44) shows that it is composed of differences of the integers s^v_a, Q, and d^v_a, and hence ϕ^v_{a+1} must be an integer for all $a \in \{1,\ldots,n^v\}$. For $\Delta^{v,2}_a$, based on (37b), its integrality depends on $\Delta^{v,2}_{a-1}$ as other variables inside the maximum and minimum operations are integers. This reclusive relationship implies that the integrality condition purely depends on $\Delta^{v,2}_0$, which is zero (which is, in turn, an integer) by definition. Hence, $\Delta^{v,2}_a$ is an integer for all $a \in \{1,\ldots,n^v\}$. All five terms in the minimization bracket in (41a) are the results of the addition, subtraction or minimization operations of five integers, s_a^v, Q, d_a^v, ϕ_{a+1}^v, and $\Delta_{a-2}^{v,2}$. Therefore, $p_a^{v,2}$ must be an integer.

Based on the conclusions of the third and fourth steps, we can conclude that $p_a^{v,2}$ must be a non-negative integer as required by (38).

To show that (42) satisfies (38), we need to show that $r_a^{v,2}$ is the result of the addition and minimization operations of the non-negative integers. In the right-hand side of (42), $d_a^{v,2}$, $p_a^{v,2}$, and $\Delta_{a-1}^{v,2}$ are non-negative integers. Therefore, $r_a^{v,2}$ must be a non-negative integer and satisfy (38).

As equations (41) and (42) satisfy constraints (29)–(30) and (36)–(38), they are feasible solutions to the loading and unloading sub-problem. This completes the proof. ■

Appendix A.4: Proof of Proposition 4

Let dd_1 and dd_2 be the minimum total demand dissatisfaction computed based on the loading/unloading strategies (39)-(40) (in Proposition 1) and (41)-(42) (in Proposition 4), respectively. As mentioned in Section 3.2, μ in the objective function can be rewritten and transformed into $\max\left(\sum_{v\in V}\sum_{j\in R\cap J_v}\left(d_j - r_{jv}\right) - TDD, 0\right)$ based on inequalities (2) and (3). From this resultant equation, with $TDD = 0$, dd_1 can be expressed as follows:

$$dd_1 = \max\left[\sum_{v\in V}\sum_{a=1}^{n^v}\left(d_a^v - r_a^{v,1}\right), 0\right]$$

$$= \sum_{v\in V}\sum_{a=1}^{n^v}\left(d_a^v - r_a^{v,1}\right) = \sum_{v\in V}\sum_{a=1}^{n^v}\left(d_a^v - \min\left(d_a^v, p_a^{v,1} + \Delta_{a-1}^{v,1}\right)\right)$$

$$= \sum_{v\in V}\sum_{a=1}^{n^v}\left(\max\left(0, d_a^v - p_a^{v,1} - \Delta_{a-1}^{v,1}\right)\right)$$

$$= \sum_{v\in V}\sum_{a=1}^{n^v}\left(\max\left(0, d_a^v - \min\left(s_a^v, Q - \Delta_{a-1}^{v,1}\right) - \Delta_{a-1}^{v,1}\right)\right)$$

$$= \sum_{v\in V}\sum_{a=1}^{n^v}\left(\max\left(0, \max(d_a^v - \Delta_{a-1}^{v,1} - s_a^v, d_a^v - Q)\right)\right)$$

$$= \sum_{v\in V}\sum_{a=1}^{n^v}\left(\max\left(0, d_a^v - \Delta_{a-1}^{v,1} - s_a^v, d_a^v - Q\right)\right).$$

For dd_2, a similar expression can be obtained:

$$dd_2 = \sum_{v\in V}\sum_{a=1}^{n^v}\left(d_a^v - r_a^{v,2}\right) = \sum_{v\in V}\sum_{a=1}^{n^v}\left(\max(0, d_a^v - p_a^{v,2} - \Delta_{a-1}^{v,2}\right)$$
$$= \sum_{v\in V}\sum_{a=1}^{n^v}\left(\max(0, d_a^v - \min\left(s_a^v, Q - \Delta_{a-1}^{v,2}, d_a^v + \phi_{a+1}^v - \Delta_{a-1}^{v,2}\right) - \Delta_{a-1}^{v,2}\right)$$
$$= \sum_{v\in V}\sum_{a=1}^{n^v}\left(\max\left(0, d_a^v - s_a^v - \Delta_{a-1}^{v,2}, d_a^v - Q, -\phi_{a+1}^v\right)\right).$$

As the fourth term in dd_2, $-\phi_{a+1}^v$, must not be greater than 0 for any value of *a*, dd_2 can be rewritten as

$$dd_2 = \sum_{v\in V}\sum_{a=1}^{n^v}\left(\max\left(0, d_a^v - s_a^v - \Delta_{a-1}^{v,2}, d_a^v - Q\right)\right).$$

Let $dd_{1,a}^v$ and $dd_{2,a}^v$ be the demand dissatisfactions at artificial drop-off node *a* in the route of vehicle *v* determined by the strategies in (39)-(40) and (41)-(42), respectively, which can be expressed as

$$dd_{1,a}^v = \max\left(0, d_a^v - s_a^v - \Delta_{a-1}^{v,1}, d_a^v - Q\right), \text{ and} \tag{51}$$
$$dd_{2,a}^v = \max\left(0, d_a^v - s_a^v - \Delta_{a-1}^{v,2}, d_a^v - Q\right).$$

The differences between loading and unloading activities at the *a*-th artificial pickup and drop-off nodes respectively in the route of vehicle *v* based on (39)-(40) and (41)-(42) can be expressed as follows:

$$p_a^{v,1} - r_a^{v,1} = p_a^{v,1} - \min\left(d_a^v, p_a^{v,1} + \Delta_{a-1}^{v,1}\right)$$
$$= \max\left[p_a^{v,1} - d_a^v, p_a^{v,1} - \left(p_a^{v,1} + \Delta_{a-1}^{v,1}\right)\right]$$
$$= \max\left[\min\left(s_a^v, Q - \Delta_{a-1}^{v,1}\right) - d_a^v, -\Delta_{a-1}^{v,1}\right].$$
$$p_a^{v,2} - r_a^{v,2} = \max\left[\min\left(s_a^v, Q - \Delta_{a-1}^{v,2}, d_a^v + \phi_{a+1}^v - \Delta_{a-1}^{v,2}\right) - d_a^v, -\Delta_{a-1}^{v,2}\right].$$

We let, along the route of vehicle *v*, artificial pickup node $m \in \{1, \ldots, n^v\}$ be the *first* artificial pickup node such that the term $d_m^v + \phi_{m+1}^v - \Delta_{m-1}^{v,2}$ is the sole minimum among $s_m^v, Q - \Delta_{m-1}^{v,2}$, and $d_m^v + \phi_{m+1}^v - \Delta_{m-1}^{v,2}$, and define artificial pickup node $m' \in \{2, \ldots, n^v\}, m' > m$ to be the *first* artificial pickup node after artificial pickup node *m* such that $s_{m'}^v + \Delta_{m'-1}^{v,2} > Q$. We also let artificial pickup node *k* be the first artificial pickup node after artificial pickup node *m'* $(\text{i.e.,} k > m')$ such that $d_k^v + \phi_{k+1}^v - \Delta_{k-1}^{v,2}$ is the sole minimum of $p_k^{v,2}$ in (41) and let artificial pickup node *k'* be the first artificial pickup node after artificial pickup node *k* $(\text{i.e.,} k' > k)$ such that $s_{k'}^v + \Delta_{k'-1}^{v,2} > Q$. We further let *l* be the first artificial pickup node after artificial pickup node *k'* $(\text{i.e.,} l > k')$ such that $d_l^v + \phi_{l+1}^v - \Delta_{l-1}^{v,2}$ is the sole minimum in $p_l^{v,2}$.

We consider the following 6 cases.

Case 1: Artificial pickup node *m* does not exist.

Case 2: Artificial pickup node *m* exists but artificial pickup node *m'* does not.

Case 3: Both artificial pickup nodes *m* and *m'* exist, but artificial pickup node *k* does not.

Case 4: Artificial pickup nodes *m*, *m'*, and *k* exist but artificial pickup node *k'* does not.
Case 5: Artificial pickup nodes *m*, *m'*, *k*, and *k'* exist but artificial pickup node *l* does not.
Case 6: Generalization of Cases 4 and 5.

Case 1: Artificial pickup node m does not exist.

In this case, for all artificial pickup nodes *a* along the route of vehicle *v*, the term $d_a^v + \phi_{a+1}^v - \Delta_{a-1}^{v,2}$ is never the sole minimum among the three terms of the minimum operator in (41). We have

$p_a^{v,2} - r_a^{v,2} = \max\left[\min\left(s_a^v, Q - \Delta_{a-1}^{v,2}\right) - d_a^v, -\Delta_{a-1}^{v,2}\right]$, and

$\Delta_a^{v,2} = p_a^{v,2} - r_a^{v,2} + \Delta_{a-1}^{v,2} = \max\left[\min\left(s_a^v + \Delta_{a-1}^{v,2}, Q\right) - d_a^v, 0\right], \forall 1 \le a \le n^v.$

Meanwhile, we have

$p_a^{v,1} - r_a^{v,1} = \max\left[\min\left(s_a^v, Q - \Delta_{a-1}^{v,1}\right) - d_a^v, -\Delta_{a-1}^{v,1}\right]$ and

$$\Delta_a^{v,1} = \max\left[\min\left(s_a^v + \Delta_{a-1}^{v,1}, Q\right) - d_a^v, 0\right], \ \forall 1 \le a \le n^v. \tag{52}$$

As $\Delta_0^{v,1} = \Delta_0^{v,2} = 0$ by definition,

$p_a^{v,1} - r_a^{v,1} = p_a^{v,2} - r_a^{v,2}, \ \forall 1 \le a \le n^v$, and therefore

$$\Delta_a^{v,1} = \Delta_a^{v,2}. \tag{53}$$

From (53), we can conclude that in the route of vehicle *v*, the demand dissatisfaction at an artificial pickup node *a* computed based on (39)-(40) is, therefore, equal to that based on (41)-(42), because

$$dd_{1,a}^v = \max\left(0, d_a^v - \Delta_{a-1}^{v,1} - s_a^v, d_a^v - Q\right) = \max\left(0, d_a^v - \Delta_{a-1}^{v,2} - s_a^v, d_a^v - Q\right) = dd_{2,a}^v. \tag{54}$$

Equation (54) implies

$$\sum_{a=1}^{n^v} dd_{1,a}^v = \sum_{a=1}^{n^v} dd_{2,a}^v.$$

This completes the proof of Case 1.

Case 2: Artificial pickup node m exists but artificial pickup node m' does not.

From Case 1, based on (53), it can be deduced that, before artificial pickup node *m*, the total demand dissatisfaction computed based on loading and unloading strategies (39)-(40) is equal to that based on (41)-(42), and therefore $\sum_{a=1}^{m-1} dd_{1,a}^v = \sum_{a=1}^{m-1} dd_{2,a}^v$.

At artificial pickup node *m* in the route of vehicle *v*, based on the loading strategy (41), $d_m^v + \phi_{m+1}^v - \Delta_{m-1}^{v,2}$ is the first time to be sole minimum among the three terms in the minimum operator in (41), i.e., $p_m^{v,2} = d_m^v + \phi_{m+1}^v - \Delta_{m-1}^{v,2}$. We have

$$\min\left(s_m^v, Q - \Delta_{m-1}^{v,2}\right) > d_m^v + \phi_{m+1}^v - \Delta_{m-1}^{v,2}, \tag{55}$$

$$\text{or } -\phi_{m+1}^v > d_m^v - \min\left(s_m^v + \Delta_{m-1}^{v,2}, Q\right). \tag{56}$$

The difference between loading and unloading quantities at artificial drop-off and pickup node *m* is

$$p_m^{v,2} - r_m^{v,2} = \max\left[\min\left(s_m^v, Q - \Delta_{m-1}^{v,2}, d_m^v + \phi_{m+1}^v - \Delta_{m-1}^{v,2}\right) - d_m^v, -\Delta_{m-1}^{v,2}\right]$$
$$= \max\left[d_m^v + \phi_{m+1}^v - \Delta_{m-1}^{v,2} - d_m^v, -\Delta_{m-1}^{v,2}\right](\because (46))$$
$$= \phi_{m+1}^v - \Delta_{m-1}^{v,2}.$$

Therefore, $\Delta_m^{v,2} = \Delta_{m-1}^{v,2} + \left(p_m^{v,2} - r_m^{v,2}\right) = \phi_{m+1}^v.$ (57)

Meanwhile, the demand dissatisfaction at artificial drop-off node m computed based on loading and unloading strategies (41)-(42) is

$$dd_{2,m}^v = \max\left(0, d_m^v - \Delta_{m-1}^{v,2} - s_m^v, d_m^v - Q\right)$$
$$= \max\left[0, d_m^v - \min\left(s_m^v + \Delta_{m-1}^{v,2}, Q\right)\right]$$
$$= 0 (\because (56)).$$

We now consider $p_m^{v,1} - r_m^{v,1}$ and $dd_{1,m}^v$. From (53) and (55), we have

$$\min\left(s_m^v, Q - \Delta_{m-1}^{v,1}\right) > d_m^v + \phi_{m+1}^v - \Delta_{m-1}^{v,1} \quad (58)$$

and hence $\min\left(s_m^v, Q - \Delta_{m-1}^{v,1}\right) - d_m^v > \phi_{m+1}^v - \Delta_{m-1}^{v,1} > -\Delta_{m-1}^{v,1}.$

Therefore, $p_m^{v,1} - r_m^{v,1}$, or equivalently $\max\left[\min\left(s_m^v, Q - \Delta_{m-1}^{v,1}\right) - d_m^v, -\Delta_{m-1}^{v,1}\right]$, can be simplified to $\min\left(s_m^v, Q - \Delta_{m-1}^{v,1}\right) - d_m^v$. That is,

$$p_m^{v,1} - r_m^{v,1} = \max\left[\min\left(s_m^v, Q - \Delta_{m-1}^{v,1}\right) - d_m^v, -\Delta_{m-1}^{v,1}\right] = \min\left(s_m^v, Q - \Delta_{m-1}^{v,1}\right) - d_m^v.$$

Meanwhile, from (53) and (56), we have

$$d_m^v - \min\left(s_m^v + \Delta_{m-1}^{v,1}, Q\right) < -\phi_{m+1}^v < 0. \quad (59)$$

Hence, the demand dissatisfaction at artificial drop-off node m computed based on loading and unloading strategies (39)-(40) is

$$dd_{1,m}^v = \max\left(0, d_m^v - s_m^v - \Delta_{m-1}^{v,1}, d_m^v - Q\right)$$
$$= \max\left[0, d_m^v - \min\left(s_m^v + \Delta_{m-1}^{v,1}, Q\right)\right] = 0,$$

where the last equality is due to (59).

$\therefore dd_{1,m}^v = dd_{2,m}^v = 0$, and the demand dissatisfactions at artificial drop-off node m computed based on the two loading and unloading strategies are equal.

At artificial drop-off node m, by definition,

$$\Delta_m^{v,1} = \left(p_m^{v,1} - r_m^{v,1}\right) + \Delta_{m-1}^{v,1}$$
$$= \min\left(s_m^v, Q - \Delta_{m-1}^{v,1}\right) - d_m^v + \Delta_{m-1}^{v,1}$$
$$> \left(d_m^v + \phi_{m+1}^v - \Delta_{m-1}^{v,1}\right) - d_m^v + \Delta_{m-1}^{v,1} = \phi_{m+1}^v = \Delta_m^{v,2}.$$

The inequality is due to (58) and the last equality is due to (57).

$$\therefore \Delta_m^{v,1} > \Delta_m^{v,2}. \quad (60)$$

Without the existence of artificial pickup node m', for any artificial pickup node b after artificial pickup node m, i.e., $\forall b \in \{m+1,\ldots,n^v\}$, by definition, we have

$$s_b^v + \Delta_{b-1}^{v,2} \leq Q \,. \tag{61}$$

Moreover, by Lemma 1 in Appendix B, we have

$$\Delta_{b-1}^{v,2} = \phi_b^v,\ \ \forall b \in \{m+1,\ldots,n^v\}. \tag{62}$$

Furthermore, by Lemma 2 in Appendix C, we have

$$\Delta_b^{v,1} \geq \Delta_b^{v,2},\ \ \forall b \in \{m,\ldots,n^v\}. \tag{63}$$

The total demand dissatisfaction after artificial drop-off node m computed based on strategies (41)-(42) is

$$\sum_{b=m+1}^{n^v} dd_{2,b}^v = \sum_{b=m+1}^{n^v} \max\left(0, d_b^v - \Delta_{b-1}^{v,2} - s_b^v, d_b^v - Q\right)$$

$$= \sum_{b=m+1}^{n^v} \max\left(0, d_b^v - \min\left(\Delta_{b-1}^{v,2} + s_b^v, Q\right)\right)$$

$$= \sum_{b=m+1}^{n^v} \max\left(0, d_b^v - \Delta_{b-1}^{v,2} - s_b^v\right) \ \left(\because s_b^v + \Delta_{b-1}^{v,2} \leq Q\right)$$

$$= \sum_{b=m+1}^{n^v} \max\left(0, d_b^v - \phi_b^v - s_b^v\right) \ (\because \text{(62)})$$

$$= \sum_{b=m+1}^{n^v} \max\left[0, d_b^v - s_b^v - \max\left(0, d_b^v - Q, d_b^v - s_b^v\right) - \phi_{b+1}^v\right]$$

$$= \sum_{b=m+1}^{n^v} \max\left[0, \min\left(d_b^v - s_b^v, Q - s_b^v, 0\right) - \phi_{b+1}^v\right] = 0 \ \left(\because \min\left(d_b^v - s_b^v, Q - s_b^v, 0\right) \leq 0, -\phi_{b+1}^v \leq 0\right).$$

Using similar techniques in deducing $\sum_{b=m+1}^{n^v} dd_{2,b}^v$, we can deduce $dd_{2,b}^v$ or equivalently

$$\begin{aligned} &\max\left(d_b^v - \min\left(s_b^v + \Delta_{b-1}^{v,2}, Q\right), 0\right) \\ &= \max\left[0, \min\left(d_b^v - s_b^v, Q - s_b^v, 0\right) - \phi_{b+1}^v\right] = 0. \end{aligned} \tag{64}$$

From (63), we have $\Delta_b^{v,1} \geq \Delta_b^{v,2}$, and therefore

$$\min\left(s_b^v + \Delta_{b-1}^{v,1}, Q\right) \geq \min\left(s_b^v + \Delta_{b-1}^{v,2}, Q\right)$$

$$d_b^v - \min\left(s_b^v + \Delta_{b-1}^{v,1}, Q\right) \leq d_b^v - \min\left(s_b^v + \Delta_{b-1}^{v,2}, Q\right)$$

$$\max\left(d_b^v - \min\left(s_b^v + \Delta_{b-1}^{v,1}, Q\right), 0\right) \leq \max\left(d_b^v - \min\left(s_b^v + \Delta_{b-1}^{v,2}, Q\right), 0\right).$$

The right side is equal to zero according to (64).

$\therefore \max\left(d_b^v - \min\left(s_b^v + \Delta_{b-1}^{v,1}, Q\right), 0\right) \leq \max\left(d_b^v - \min\left(s_b^v + \Delta_{b-1}^{v,2}, Q\right), 0\right) = 0$, which implies that

$$\max\left(d_b^v - \min\left(s_b^v + \Delta_{b-1}^{v,1}, Q\right), 0\right) = 0. \tag{65}$$

The total demand dissatisfaction computed based on the loading strategies from Proposition 1 becomes

$$\sum_{b=m+1}^{n^v} dd_{1,b}^v = \sum_{b=m+1}^{n^v} \max\left(0, d_b^v - \Delta_{b-1}^{v,1} - s_b^v, d_b^v - Q\right)$$

$$= \sum_{b=m+1}^{n^v} \max\left(0, d_b^v - \min\left(\Delta_{b-1}^{v,1} + s_b^v, Q\right)\right) = 0 \text{ } (\because (65)),$$

which is equal to the one based on Proposition 4.

The above proofs show that the total demand dissatisfactions before artificial drop-off node *m*, at drop-off artificial drop-off node *m*, and after artificial drop-off node *m* are equal by adopting the loading strategies of Proposition 1 or Proposition 4. Mathematically, $\sum_{a=1}^{n^v} dd_{1,a}^v = \sum_{a=1}^{n^v} dd_{2,a}^v$. This completes the proof of Case 2.

Case 3: Both artificial pickup nodes m and m' exist, but artificial pickup node k does not.

Case 2 shows that the total demand dissatisfactions before and at artificial drop-off node *m* computed based on loading strategies by Proposition 1 are equal to those based on (41). Case 2 also shows that the total demand dissatisfactions from artificial drop-off node (*m*+1) to artificial drop-off node (*m'*-1) are equal based on the strategies of both propositions. In other words,

$$\sum_{a=1}^{m'-1} dd_{1,a}^v = \sum_{a=1}^{m'-1} dd_{2,a}^v. \tag{66}$$

At artificial pickup node *m'* of the route of vehicle *v*, we have $s_{m'}^v + \Delta_{m'-1}^{v,2} > Q$ by definition. The demand dissatisfaction at artificial drop-off node *m'* computed based on Proposition 4 is

$$dd_{2,m'}^v = \max\left(d_{m'}^v - s_{m'}^v - \Delta_{m'-1}^{v,2}, d_{m'}^v - Q, 0\right) = \max\left(d_{m'}^v - Q, 0\right). \tag{67}$$

From (63) and the definition $s_{m'}^v + \Delta_{m'-1}^{v,2} > Q$, we have

$$s_{m'}^v + \Delta_{m'-1}^{v,1} \geq s_{m'}^v + \Delta_{m'-1}^{v,2} > Q. \tag{68}$$

Meanwhile, the demand dissatisfaction at artificial drop-off node *m'* computed based on the strategies of Proposition 1 is

$$dd_{1,m'}^v = \max\left(d_{m'}^v - s_{m'}^v - \Delta_{m'-1}^{v,1}, d_{m'}^v - Q, 0\right)$$

$$= \max\left(d_{m'}^v - \min\left(s_{m'}^v + \Delta_{m'-1}^{v,1}, Q\right), 0\right) = \max\left(d_{m'}^v - Q, 0\right).$$

$$\therefore dd_{1,m'}^v = dd_{2,m'}^v = \max\left(d_{m'}^v - Q, 0\right). \tag{69}$$

The number of bikes leaving artificial drop-off node *m'* computed based on the loading and unloading strategies of Proposition 4 is

$$\Delta_{m'}^{v,2} = \Delta_{m'-1}^{v,2} + p_{m'}^{v,2} - r_{m'}^{v,2}$$

$$= \Delta_{m'-1}^{v,2} + \max\left[\min\left(s_{m'}^v, Q - \Delta_{m'-1}^{v,2}, d_{m'}^v + \phi_{m'+1}^v - \Delta_{m'-1}^{v,2}\right) - d_{m'}^v, -\Delta_{m'-1}^{v,2}\right]$$

$$= \max\left[\min\left(s_{m'}^v + \Delta_{m'-1}^{v,2} - d_{m'}^v, Q - d_{m'}^v, \phi_{m'+1}^v\right), 0\right]$$

$$= \max\left[\min\left(Q - d_{m'}^v, \phi_{m'+1}^v\right), 0\right]. \text{ } (\because s_{m'}^v + \Delta_{m'-1}^{v,2} > Q \text{, i.e., based on the definition of node } m')$$

Based on (67), there are only two possible values of $dd^v_{2,m'}$, $d^v_{m'}-Q$, and 0. When $dd^v_{2,m'}=d^v_{m'}-Q>0$, $Q-d^v_{m'}<0$ and therefore $\Delta^{v,2}_{m'}=0$; if $dd^v_{2,m'}=0$, $Q-d^v_{m'}\geq 0$ and therefore $\Delta^{v,2}_{m'}=\min\left(Q-d^v_{m'},\phi^v_{m'+1}\right)$.
Meanwhile, the number of bikes leaving artificial drop-off node *m'* based on Proposition 1 is

$$\Delta^{v,1}_{m'}=\Delta^{v,1}_{m'-1}+p^{v,1}_{m'}-r^{v,1}_{m'}$$
$$=\Delta^{v,1}_{m'-1}+\max\left[\min\left(s^v_{m'},Q-\Delta^{v,1}_{m'-1}\right)-d^{v,1}_{m'},-\Delta^{v,1}_{m'-1}\right]$$
$$=\max\left[\min\left(s^v_{m'}+\Delta^{v,1}_{m'-1}-d^v_{m'},Q-d^v_{m'}\right),0\right]=\max\left[Q-d^v_{m'},0\right]\ (\because(68)).$$

Based on (69), there are only two possible values of $dd^v_{1,m'}$, $d^v_{m'}-Q$, and 0. When $dd^v_{1,m'}=d^v_{m'}-Q>0$, $\Delta^{v,1}_{m'}=0$; when $dd^v_{1,m'}=0$, $\Delta^{v,1}_{m'}=Q-d^v_{m'}$. To sum up, when $dd^v_{1,m'}=dd^v_{2,m'}=d^v_{m'}-Q$, we have $\Delta^{v,1}_{m'}=\Delta^{v,2}_{m'}=0$; when $dd^v_{1,m'}=dd^v_{2,m'}=0$, we have $\Delta^{v,1}_{m'}=Q-d^v_{m'}$ and $\Delta^{v,2}_{m'}=\min\left(Q-d^v_{m'},\phi_{m'+1}\right)$, and therefore we can conclude $\Delta^{v,1}_{m'}\geq\Delta^{v,2}_{m'}$.

Table 7 Possible instances when vehicle *v* leaves from artificial drop-off node *m'*

Instance	Description	$\Delta^{v,1}_{m'}$	$\Delta^{v,2}_{m'}$
1	$dd^v_{1,m'}=dd^v_{2,m'}=0$	$Q-d^v_{m'}$	$\min\left(Q-d^v_{m'},\phi_{m'+1}\right)$
2	$dd^v_{1,m'}=dd^v_{2,m'}=d^v_{m'}-Q$	0	0

Table 7 lists all instances of the number of bikes on vehicle *v* leaving artificial drop-off node *m'* according to the demand dissatisfaction at artificial drop-off node *m'*. Given that there is *no* artificial pickup node *k* after artificial pickup node *m'* such that $d^v_k+\phi^v_{k+1}-\Delta^{v,2}_{k-1}$ is the sole minimum in $p^{v,2}_k$, when $dd^v_{1,m'}=dd^v_{2,m'}=d^v_{m'}-Q$, the total demand dissatisfaction after artificial drop-off node *m'* computed based on the strategies of Proposition 4 is

$$\sum_{b=m'+1}^{n^v}dd^v_{2,b}=\sum_{b=m'+1}^{n^v}\max\left(0,d^v_b-\Delta^{v,2}_{b-1}-s^v_b,d^v_b-Q\right),$$

while the total demand dissatisfaction after artificial drop-off node *n* computed based on the strategies of Proposition 1 is

$$\sum_{b=m'+1}^{n^v}dd^v_{1,b}=\sum_{b=m'+1}^{n^v}\max\left(0,d^v_b-\Delta^{v,1}_{b-1}-s^v_b,d^v_b-Q\right).$$

We also have

$$p^{v,2}_b-r^{v,2}_b=\max\left[\min\left(s^v_b,Q-\Delta^{v,2}_{b-1}\right)-d^v_b,-\Delta^{v,2}_{b-1}\right],\text{ and}$$
$$\Delta^{v,2}_b=p^{v,2}_b-r^{v,2}_b+\Delta^{v,2}_{b-1}=\max\left[\min\left(s^v_b+\Delta^{v,2}_{b-1},Q\right)-d^v_b,0\right],\ \forall m'\leq b\leq n^v.$$

Meanwhile, we have

$$p^{v,1}_b-r^{v,1}_b=\max\left[\min\left(s^v_b,Q-\Delta^{v,1}_{b-1}\right)-d^v_b,-\Delta^{v,1}_{b-1}\right]\text{ and}$$
$$\Delta^{v,1}_b=\max\left[\min\left(s^v_b+\Delta^{v,1}_{b-1},Q\right)-d^v_b,0\right],\ \forall m'\leq b\leq n^v.$$

From instance 2, we have $\Delta^{v,1}_{m'}=\Delta^{v,2}_{m'}=0$, and

$p_b^{v,1} - r_b^{v,1} = p_b^{v,2} - r_b^{v,2},\ \forall m'+1 \le b \le n^v$, and therefore

$$\Delta_b^{v,1} = \Delta_b^{v,2},\ \forall m' \le b \le n^v. \tag{70}$$

From (70), we can conclude that in the route of vehicle v, the demand dissatisfaction at an artificial drop-off node b after artificial drop-off node m' computed based on (39)-(40) is therefore equal to that based on (41)-(42), because

$$dd_{1,b}^v = \max\left(0, d_b^v - \Delta_{b-1}^{v,1} - s_b^v, d_b^v - Q\right) = \max\left(0, d_b^v - \Delta_{b-1}^{v,2} - s_b^v, d_b^v - Q\right) = dd_{2,b}^v. \tag{71}$$

Equation (71) implies

$$\sum_{b=m'+1}^{n^v} dd_{1,b}^v = \sum_{b=m'+1}^{n^v} dd_{2,b}^v, \tag{72}$$

and therefore $\sum_{a=1}^{n^v} dd_{1,a}^v = \sum_{a=1}^{n^v} dd_{2,a}^v$ for instance 2.

From instance 1, when $dd_{1,m'}^v = dd_{2,m'}^v = 0$ and $Q - d_{m'}^v < \phi_{m'+1}$, we have $\Delta_{m'}^{v,1} = \Delta_{m'}^{v,2} = Q - d_{m'}^v$. Replacing $\Delta_{m'}^{v,1} = \Delta_{m'}^{v,2} = 0$ with $\Delta_{m'}^{v,1} = \Delta_{m'}^{v,2} = Q - d_{m'}^v$, the steps that are used to deduce equations (70) and (71) can be applied here to show $dd_{1,b}^v = dd_{2,b}^v,\ \forall m'+1 \le b \le n^v$ because $\Delta_{m'}^{v,1} = \Delta_{m'}^{v,2}$. This implies that

$$\sum_{b=m'+1}^{n^v} dd_{1,b}^v = \sum_{b=m'+1}^{n^v} dd_{2,b}^v \tag{73}$$

and therefore $\sum_{a=1}^{n^v} dd_{1,a}^v = \sum_{a=1}^{n^v} dd_{2,a}^v$ when $\Delta_{m'}^{v,1} = \Delta_{m'}^{v,2} = Q - d_{m'}^v$.

From instance 1, when $dd_{1,m'}^v = dd_{2,m'}^v = 0$ and $Q - d_{m'}^v > \phi_{m'+1}$, we have $\Delta_{m'}^{v,1} = Q - d_{m'}^v \ge \phi_{m'+1}^v = \Delta_{m'}^{v,2}$. The total demand dissatisfaction after artificial drop-off node m' computed based on the strategies of Proposition 4 is

$$\sum_{b=m'+1}^{n^v} dd_{2,b}^v = \sum_{b=m'+1}^{n^v} \max\left(0, d_b^v - \Delta_{b-1}^{v,2} - s_b^v, d_b^v - Q\right)$$

$$= \sum_{b=m'+1}^{n^v} \max\left(0, d_b^v - \min\left(\Delta_{b-1}^{v,2} + s_b^v, Q\right)\right)$$

$$= \sum_{b=m'+1}^{n^v} \max\left(0, d_b^v - \Delta_{b-1}^{v,2} - s_b^v\right) \quad \left(\because s_b^v + \Delta_{b-1}^{v,2} \le Q\right)$$

$$= \sum_{b=m'+1}^{n^v} \max\left(0, d_b^v - \phi_b^v - s_b^v\right) \ (\because (62))$$

$$= \sum_{b=m'+1}^{n^v} \max\left[0, d_b^v - s_b^v - \max\left(0, d_b^v - Q, d_b^v - s_b^v\right) - \phi_{b+1}^v\right]$$

$$= \sum_{b=m'+1}^{n^v} \max\left[0, \min\left(d_b^v - s_b^v, Q - s_b^v, 0\right) - \phi_{b+1}^v\right] = 0 \ (\because \min\left(d_b^v - s_b^v, Q - s_b^v, 0\right) \le 0, -\phi_{b+1}^v \le 0).$$

Using similar techniques in deducing $\sum_{b=m'+1}^{n^v} dd_{2,b}^v$, we can deduce $dd_{2,b}^v$ or equivalently

$$\max\left(d_b^v - \min\left(s_b^v + \Delta_{b-1}^{v,2}, Q\right), 0\right)$$
$$= \max\left[0, \min\left(d_b^v - s_b^v, Q - s_b^v, 0\right) - \phi_{b+1}^v\right] = 0. \tag{74}$$

Meanwhile, the above paragraph has shown that $\Delta_{m'}^{v,1} \geq \Delta_{m'}^{v,2}$, and by (63), $\Delta_b^{v,1} \geq \Delta_b^{v,2},\ \forall b \in \left\{m'+1,\ldots,n^v\right\}$. We have

$$\min\left(s_b^v + \Delta_{b-1}^{v,1}, Q\right) \geq \min\left(s_b^v + \Delta_{b-1}^{v,2}, Q\right)$$
$$d_b^v - \min\left(s_b^v + \Delta_{b-1}^{v,1}, Q\right) \leq d_b^v - \min\left(s_b^v + \Delta_{b-1}^{v,2}, Q\right)$$
$$\max\left(d_b^v - \min\left(s_b^v + \Delta_{b-1}^{v,1}, Q\right), 0\right) \leq \max\left(d_b^v - \min\left(s_b^v + \Delta_{b-1}^{v,2}, Q\right), 0\right).$$

The right side is equal to zero according to (74).

$\therefore \max\left(d_b^v - \min\left(s_b^v + \Delta_{b-1}^{v,1}, Q\right), 0\right) \leq \max\left(d_b^v - \min\left(s_b^v + \Delta_{b-1}^{v,2}, Q\right), 0\right) = 0$, which implies that

$$\max\left(d_b^v - \min\left(s_b^v + \Delta_{b-1}^{v,1}, Q\right), 0\right) = 0. \tag{75}$$

The total demand dissatisfaction computed based on the strategies in equations (39)-(40) can, therefore, be expressed as

$$\sum_{b=m'+1}^{n^v} dd_{1,b}^v = \sum_{b=m'+1}^{n^v} \max\left(0, d_b^v - \Delta_{b-1}^{v,1} - s_b^v, d_b^v - Q\right)$$
$$= \sum_{b=m'+1}^{n^v} \max\left(0, d_b^v - \min\left(\Delta_{b-1}^{v,1} + s_b^v, Q\right)\right) = 0 \ (\because (75)),$$

which is equal to the one based on equations (41)-(42), i.e.,

$$\sum_{b=m'+1}^{n^v} dd_{1,b}^v = \sum_{b=m'+1}^{n^v} dd_{2,b}^v. \tag{76}$$

and therefore $\sum_{a=1}^{n^v} dd_{1,a}^v = \sum_{a=1}^{n^v} dd_{2,a}^v$ when $\Delta_{m'}^{v,1} = Q - d_{m'}^v$ and $\Delta_{m'}^{v,2} = \phi_{m'+1}^v$.

To sum up, from equations (72), (73), and (76), in all instances, we have

$$\sum_{b=m'+1}^{n^v} dd_{1,b}^v = \sum_{b=m'+1}^{n^v} dd_{2,b}^v. \tag{77}$$

The above proofs show that the total demand dissatisfactions before artificial drop-off node *m'*, at artificial drop-off node *m'*, and after artificial drop-off node *m'* are equal by adopting the strategies based on (39)-(40) and (41)-(42). Mathematically, $\sum_{a=1}^{n^v} dd_{1,a}^v = \sum_{a=1}^{n^v} dd_{2,a}^v$. This completes the proof of Case 3.

Case 4: Artificial pickup nodes m, m', and k exist but artificial pickup node k' does not.

From Case 3, $\sum_{a=1}^{m'} dd_{1,a}^v = \sum_{a=1}^{m'} dd_{2,a}^v$. Based on the logic to deduce (77), from artificial drop-off node (*m'*+1) to artificial drop-off node (*k*-1), the total demand dissatisfaction computed based on loading and unloading strategies (39)-(40) is, therefore, equal to that based on (41)-(42), and therefore

$$\sum_{a=m'+1}^{k-1} dd_{1,a}^{v} = \sum_{a=m'+1}^{k-1} dd_{2,a}^{v}.$$

By adopting the loading strategy (41) at artificial pickup node *k*, and by (57), we have

$$\Delta_k^{v,2} = \phi_{k+1}^{v}. \tag{78}$$

Moreover, from (60), we have

$$\Delta_k^{v,1} > \Delta_k^{v,2}. \tag{79}$$

Meanwhile, by replacing *m* with *k*, we follow the approach to deduce (55)-(59) to determine the demand dissatisfaction at artificial drop-off node *k*. It shows the demand dissatisfaction at artificial drop-off node *k* computed based on the loading and unloading strategies (39)-(40) is equal to that based on (41)-(42), i.e., $dd_{1,k}^{v} = dd_{2,k}^{v}$.

For all artificial nodes after artificial node *k*, we can follow the steps in deducing (60)-(65) to determine the total demand dissatisfaction after artificial drop-off node *k* as shown in Case 2 through replacing *m* with *k*, and therefore the total demand dissatisfaction after artificial drop-off node *k* computed based on the loading and unloading strategies (39)-(40) is therefore equal to that based on (41)-(42).

The above proofs show that the total demand dissatisfactions before, at, and after artificial drop-off node *k* are equal by adopting the loading strategies of (39)-(40) or (41)-(42). Mathematically, $\sum_{a=1}^{n^v} dd_{1,a}^{v} = \sum_{a=1}^{n^v} dd_{2,a}^{v}$. This completes the proof of Case 4.

Case 5: Artificial pickup nodes m, m', k, and k' exist but artificial pickup node l does not.

As mentioned in Case 4, $\sum_{a=1}^{m'} dd_{1,a}^{v} = \sum_{a=1}^{m'} dd_{2,a}^{v}$. From Case 4, it can be deduced that from artificial drop-off node (*k*+1) to artificial drop-off node (*k'*-1), the total demand dissatisfaction computed based on loading and unloading strategies (39)-(40) is therefore equal to that based on (41)-(42), and therefore $\sum_{a=k+1}^{k'-1} dd_{1,a}^{v} = \sum_{a=k+1}^{k'-1} dd_{2,a}^{v}$.

At artificial pickup node *k'*, we have $s_{k'}^{v} + \Delta_{k'-1}^{v,2} > Q$, which is similar to the case at artificial pickup node *m'*. The demand dissatisfaction at artificial drop-off node *k'* can be determined by replacing *m'* with *k'* in (67)-(69), which gives

$$dd_{1,k'}^{v} = dd_{2,k'}^{v} = \max\left(d_{k'}^{v} - Q, 0\right). \tag{80}$$

For the artificial pickup nodes after artificial pickup node *k'*, the total demand dissatisfaction can be determined following the steps to deduce (70)-(76). This gives the same conclusion as (77) that the total demand dissatisfaction after artificial drop-off node *k'* computed based on the loading and unloading strategies (39)-(40) is identical to the one based on (41)-(42).

Therefore, the total demand dissatisfactions before, at, and after artificial drop-off node *k'* are equal by adopting the loading and unloading strategies of (39)-(40) or (41)-(42). Mathematically, $\sum_{a=1}^{n^v} dd_{1,a}^v = \sum_{a=1}^{n^v} dd_{2,a}^v$. This completes the proof of Case 5.

Case 6: Generalization of Cases 4 and 5.

Cases 4 and 5 cover the situations for having artificial pickup node *k* after artificial pickup node *m'* with and without artificial pickup node *k'*, respectively and conclude that the total demand dissatisfactions under the strategies from Proposition 1 and Proposition 4 remain the same. In other words, we can conclude that, with two pairs of artificial pickup nodes (b'_ϖ, b''_ϖ), $\varpi = 1,2$ with $b''_\varpi > b'_\varpi$ and $b'_\varpi > b''_\varpi, \varpi = 2$ such that $d_{b'_\varpi}^v + \phi_{b'_\varpi}^v - \Delta_{b'_\varpi - 1}^{v,2}$ is the sole minimum based on (41) and $s_{b''_\varpi}^v + \Delta_{b''_\varpi - 1}^{v,2} > Q$, the total demand dissatisfactions along the whole route computed based on loading and unloading strategies of (39)-(40) and (41)-(42) are identical. Moreover, this is true even when b''_ϖ does not exist in the final pair. By using similar arguments for handling the existence of the second pair in Case 5 to each additional pair with artificial pickup node b''_ϖ and the arguments for handling the existence of the second pair in Case 4 for the last pair without b''_ϖ, we can conclude that when the number of pairs of artificial pickup nodes (b'_ϖ, b''_ϖ) is greater than two along any route, the total demand dissatisfactions calculated on the basis of the strategies in both Proposition 1 and Proposition 4 are the same. This completes the proof of Case 6.

All cases are considered. This completes the proof.■

Appendix A.5: Proof of Proposition 9

We need to prove that condition (45) satisfies $\sum_{f'=1}^{|P_a^v|} p_{a,f'}^v = p_a^{v*}$. It is clear that $p_a^{v*} - \sum_{f'=1}^{f-1} s_{a,f'}^v$ is non-increasing when *f* increases. Moreover, $p_a^{v*} - \sum_{f'=1}^{f-1} s_{a,f'}^v$ can equal 0 and $s_{a,f}^v$ as two special cases (because $p_a^{v*} \geq s_{a,f}^v$ by definition, $\sum_{f'=1}^{f} s_{a,f'}^v \geq 0$, and p_a^{v*} can equal $\sum_{f'=1}^{f} s_{a,f'}^v$ as a special case). Therefore, without loss of generality, there exists a pickup node, let it be the *e*-th node of the *a*-th artificial pickup node in the route of vehicle *v* such that $s_{a,e}^v \geq p_a^{v*} - \sum_{f'=1}^{e-1} s_{a,f'}^v \geq 0$ holds. This condition leads to the following:

- By condition (45), $p_{a,e}^v = p_a^{v*} - \sum_{f'=1}^{e-1} s_{a,f'}^v$.
- $p_a^{v*} - \sum_{f'=1}^{e-1} s_{a,f'}^v \geq 0$ is equivalent to $p_a^{v*} - \sum_{f'=1}^{e-2} s_{a,f'}^v - s_{a,e-1}^v \geq 0$ which implies $p_a^{v*} - \sum_{f'=1}^{e-2} s_{a,f'}^v \geq s_{a,e-1}^v$ and $p_a^{v*} - \sum_{f'=1}^{e-2} s_{a,f'}^v \geq 0$. Similarly, $p_a^{v*} - \sum_{f'=1}^{e-2} s_{a,f'}^v \geq 0 \Rightarrow p_a^{v*} - \sum_{f'=1}^{e-3} s_{a,f'}^v \geq s_{a,e-2}^v$ and $p_a^{v*} - \sum_{f'=1}^{e-3} s_{a,f'}^v \geq 0$.

By repeating these steps, we have $s^v_{a,f} \le p^{v*}_a - \sum_{f'=1}^{f-1} s^v_{a,f'}, 0 \le p^{v*}_a - \sum_{f'=1}^{f-1} s^v_{a,f'}$, $f = 1,\dots,e-1$, leading to $p^v_{a,f} = s^v_{a,f}, f = 1,\dots,e-1$ according to condition (45).

- $s^v_{a,e} \ge p^{v*}_a - \sum_{f'=1}^{e-1} s^v_{a,f'}$ is equivalent to $s^v_{a,e} - s^v_{a,e} \ge p^{v*}_a - \sum_{f'=1}^{e-1} s^v_{a,f'} - s^v_{a,e}$ and $0 \ge p^{v*}_a - \sum_{f'=1}^{e} s^v_{a,f'}$. $0 \ge p^{v*}_a - \sum_{f'=1}^{e} s^v_{a,f'} \Rightarrow 0 \ge p^{v*}_a - \sum_{f'=1}^{f} s^v_{a,f'}$, $f = e+1,\dots,\left|P^v_a\right|$ because $s^v_{a,f} \ge 0$, $f = e+1,\dots,\left|P^v_a\right|$. Therefore, by condition (45), $p^v_{a,f} = 0, f = e+1,\dots,\left|P^v_a\right|$.

Consequently, $\sum_{f'=1}^{\left|P^v_a\right|} p^v_{a,f'} = \sum_{f'=1}^{e-1} s^v_{a,f'} + \left(p^{v*}_a - \sum_{f'=1}^{e-1} s^v_{a,f'} \right) + \sum_{f'=e}^{\left|P^v_a\right|} 0 = p^{v*}_a$, which is consistent with the total number of bikes picked up at the a-th artificial pickup node when TDD = 0. As p^{v*}_a is optimal, any feasible allocation of this value to the pickup quantity of each node in the a-th artificial pickup node is also optimal. ■

Appendix A.6: Proof of Proposition 10

The functional form of equation (46) is the same as that of equation (45), and the functional form of $\sum_{f'=1}^{\left|P^v_a\right|} p^v_{a,f'} = p^{v*}_a$ is the same as that of $\sum_{f'=1}^{\left|R^v_a\right|} r^v_{a,f'} = r^{v*}_a$. Therefore, we can follow the proof of Proposition 9, but p^{v*}_a, $s^v_{a,f}$, $p^v_{a,f}$, $\left|P^v_a\right|$, and equation (45), are respectively replaced by r^{v*}_a, $d^v_{a,f}$, $r^v_{a,f}$, $\left|R^v_a\right|$, and equation (46). ■

Appendix A.7: Proof of Proposition 11

We mainly need to prove that $\sum_{v=1}^{|V|}\sum_{c=1}^{n^v} r^{v,opt}_c = \sum_{v=1}^{|V|}\sum_{c=1}^{n^v} r^{v*}_c - TDD$. It is clear that $\sum_{v=1}^{v'-1}\sum_{c=1}^{n^v} r^{v*}_c + \sum_{c=1}^{b} r^{v'*}_c - TDD$ is *non-decreasing* when *v'* or *b* increases. Moreover, $\sum_{v=1}^{v'-1}\sum_{c=1}^{n^v} r^{v*}_c + \sum_{c=1}^{b} r^{v'*}_c - TDD$ can become 0 and $r^{v'*}_b$ as two special cases (in which the latter is because $\sum_{v=1}^{|V|}\sum_{c=1}^{n^v} r^{v*}_c \ge r^{v'*}_b$). Without loss of generality, there exists an artificial node in a route, let it be the z-th artificial node in the route of vehicle u, such that $0 \le \sum_{v=1}^{u-1}\sum_{c=1}^{n^v} r^{v*}_c + \sum_{c=1}^{z} r^{u*}_c - TDD \le r^{u*}_z$ holds. This condition leads to the following:

- By condition (47), $r^{u,opt}_z = \sum_{v=1}^{u-1}\sum_{c=1}^{n^v} r^{v*}_c + \sum_{c=1}^{z} r^{u*}_c - TDD$.
- $\sum_{v=1}^{u-1}\sum_{c=1}^{n^v} r^{v*}_c + \sum_{c=1}^{z} r^{u*}_c - TDD \le r^{u*}_z$ is equivalent to $\sum_{v=1}^{u-1}\sum_{c=1}^{n^v} r^{v*}_c + \sum_{c=1}^{z-1} r^{u*}_c - TDD \le 0$, which implies $\sum_{v=1}^{u-1}\sum_{c=1}^{n^v} r^{v*}_c + \sum_{c=1}^{a'-1} r^{u*}_c - TDD \le 0$, $a' = 1,\dots,z$ and $\sum_{v=1}^{u'-1}\sum_{c=1}^{a''} r^{v*}_c - TDD \le 0$, $u' = 1,\dots,u-1$ and

$a''=1,\ldots,n^{u'}$, because $r_c^{v*}\geq 0,\forall v\in V,c\in\{1,\ldots,n^v\}$. Therefore, by condition (47), $r_c^{u,opt}=0$, $c=1,\ldots,z-1$ and $r_c^{u',opt}=0$, $u'=1,\ldots,u-1$, $c=1,\ldots,n^{u'}$.

- $\sum_{v=1}^{u-1}\sum_{c=1}^{n^v}r_c^{v*}+\sum_{c=1}^{z}r_c^{u*}-TDD\geq 0$ implies $\sum_{v=1}^{u-1}\sum_{c=1}^{n^v}r_c^{v*}+\sum_{c=1}^{a''}r_c^{u*}-TDD\geq 0$, $a''=z+1,\ldots,n^u$ and $\sum_{v=1}^{w'}\sum_{i=1}^{a'}r_i^{v*}-TDD\geq 0$, $w'=u+1,\ldots,|V|$, $a'=1,\ldots,n^{w'}$ because $r_c^{u*}\geq 0,c=z+1,\ldots,n^u$ and $r_c^{u'*}\geq 0$, $u'=u+1,\ldots,|V|$, $c=1,\ldots,n^{u'}$, respectively. $\sum_{v=1}^{u-1}\sum_{c=1}^{n^v}r_c^{v*}+\sum_{c=1}^{z}r_c^{u*}-TDD\geq 0$ also implies $\sum_{v=1}^{u-1}\sum_{c=1}^{n^v}r_c^{v*}+\sum_{c=1}^{a''}r_c^{u*}-TDD\geq r_{a''}^{u*}$, $a''=z+1,\ldots,n^u$ and $\sum_{v=1}^{u'}\sum_{c=1}^{a'}r_c^{v*}-TDD\geq r_{a'}^{u'*}$, $u'=u+1,\ldots,|V|$, $a'=1,\ldots,n^{u'}$ because $r_c^{u*}\geq 0,c=z+1,\ldots,n^u$ and $r_c^{u'*}\geq 0$, $u'=u+1,\ldots,|V|$, $c=1,\ldots,n^{u'}$, respectively. Therefore, by condition (47), $r_c^{u',opt}=r_c^{u'}$, $u'=u+1,\ldots,|V|$, $c=1,\ldots,n^{u'}$, and $r_c^{u,opt}=r_c^u$, $c=z+1,\ldots,n^u$.

Thus,

$$\sum_{v=1}^{|V|}\sum_{c=1}^{n^v}r_c^{v,opt}=\left(\sum_{v=1}^{u-1}\sum_{c=1}^{n^v}0+\sum_{c=1}^{z-1}0\right)+\left(\sum_{v=1}^{u-1}\sum_{c=1}^{n^v}r_i^{v*}+\sum_{c=1}^{z}r_c^{u*}-TDD\right)+\left(\sum_{c=z+1}^{n^u}r_c^{u*}+\sum_{v=u+1}^{|V|}\sum_{c=1}^{n^v}r_c^{v*}\right)=\sum_{v=1}^{|V|}\sum_{c=1}^{n^v}r_c^{v*}-TDD,$$

and the claim follows.■

Appendix A.8: Proof of Proposition 12

The functional form of equation (49) is the same as that of equation (47) and the functional form of $\sum_{v=1}^{|V|}\sum_{c=1}^{n^v}p_c^{v,opt}=\sum_{v=1}^{|V|}\sum_{c=1}^{n^v}p_c^{v*}-TDD$ is the same as $\sum_{v=1}^{|V|}\sum_{c=1}^{n^v}r_c^{v,opt}=\sum_{v=1}^{|V|}\sum_{c=1}^{n^v}r_c^{v*}-TDD$. Therefore, we can follow the proof of Proposition 11, but $r_a^{v',opt}$, $r_a^{v'*}$, $\sum_{v=1}^{v'-1}\sum_{c=1}^{n^v}r_c^{v*}+\sum_{c=1}^{a}r_c^{v'*}-TDD$, and equation (49) are respectively replaced by $p_a^{v',opt}$, $p_a^{v'*}$, $\sum_{c=1}^{a}p_c^{v*}-TDD^{v'}$, and equation (47), and also $TDD=\sum_{v=1}^{|V|}TDD^v$ is used during substituting equation (49) into $\sum_{v=1}^{|V|}\sum_{c=1}^{n^v}p_c^{v,opt}=\sum_{v=1}^{|V|}\sum_{c=1}^{n^v}p_c^{v*}-TDD$.■

APPENDIX B: PROOF OF LEMMA 1

Lemma 1 $\Delta_{b-1}^{v,2}=\phi_b^v$, $\forall b\in\{m+1,\ldots,n^v\}$.

Proof $\because \Delta_m^{v,2}=\phi_{m+1}^v$ (see (51)), and

$\Delta_{m+1}^{v,2}=\max\left[\min\left(s_{m+1}^v+\Delta_m^{v,2}-d_{m+1}^v,Q-d_{m+1}^v,\phi_{m+2}^v\right),0\right]$

$=\max\left[\min\left(s_{m+1}^v+\Delta_m^{v,2}-d_{m+1}^v,\phi_{m+2}^v\right),0\right]$ $\left(\because s_{m+1}^v+\Delta_m^{v,2}\leq Q\right)$(i.e., (61))

$$= \max\left[\min\left(s^v_{m+1} + \phi^v_{m+1} - d^v_{m+1}, \phi^v_{m+2}\right), 0\right]$$
$$= \max\left\{\min\left[s^v_{m+1} + \phi^v_{m+2} + \max\left(d^v_{m+1} - s^v_{m+1}, d^v_{m+1} - Q, 0\right) - d^v_{m+1}, \phi^v_{m+2}\right], 0\right\}$$
$$= \max\left\{\min\left[\phi^v_{m+2} + \max\left(s^v_{m+1} - Q, s^v_{m+1} - d^v_{m+1}, 0\right), \phi^v_{m+2}\right], 0\right\}$$
$$= \max\left(\phi^v_{m+2}, 0\right) = \phi^v_{m+2} \left(\because \phi^v_{m+2} \geq 0\right)$$
$$\therefore \Delta^{v,2}_{m+1} = \phi^v_{m+2}.$$

By deduction, we have $\Delta^{v,2}_{b-1} = \phi^v_b, \ \forall b \in \left\{m+1, \ldots, n^v\right\}$.■

APPENDIX C: PROOF OF LEMMA 2

Lemma 2 $\Delta^{v,1}_b \geq \Delta^{v,2}_b, \forall b \in \left\{m, \ldots, n^v\right\}$.

Proof Let $P(b)$ be the proposition that $\Delta^{v,1}_b \geq \Delta^{v,2}_b, \ \forall b \in \left\{m, \ldots, n^v\right\}$. For $b = m$, from (60), $\Delta^{v,1}_m \geq \Delta^{v,2}_m$. $\therefore P(m)$ is true.

Assume $b = \beta$, $\Delta^{v,1}_\beta \geq \Delta^{v,2}_\beta = \phi^v_{\beta+1}$ (from (62)). For $b = \beta + 1$,

$$\Delta^{v,1}_{\beta+1} - \Delta^{v,2}_{\beta+1} = \max\left[\min\left(s^v_{\beta+1} + \Delta^{v,1}_\beta, Q\right) - d^v_{\beta+1}, 0\right] - \phi^v_{\beta+2} \ (\because (52) \text{ and } (62))$$
$$\geq \max\left[\min\left(s^v_{\beta+1} + \Delta^{v,2}_\beta, Q\right) - d^v_{\beta+1}, 0\right] - \phi^v_{\beta+2} \quad \left(\because \min\left(s^v_\beta + \Delta^{v,1}_{\beta-1}, Q\right) \geq \min\left(s^v_\beta + \Delta^{v,2}_{\beta-1}, Q\right)\right)$$
$$= \max\left(s^v_{\beta+1} + \phi_{\beta+1} - d^v_{\beta+1}, 0\right) - \phi^v_{\beta+2} \qquad \left(\because s^v_\beta + \Delta^{v,2}_{\beta-1} \leq Q\right)$$
$$= \max\left(\phi^v_{\beta+2} + \max\left(0, s^v_{\beta+1} - Q, s^v_{\beta+1} - d^v_{\beta+1}\right), 0\right) - \phi^v_{\beta+2}$$
$$= \max\left(\max\left(0, s^v_{\beta+1} - Q, s^v_{\beta+1} - d^v_{\beta+1}\right), -\phi^v_{\beta+2}\right) \geq 0.$$

$\therefore \Delta^{v,1}_{\beta+1} - \Delta^{v,2}_{\beta+1} \geq 0$ and thus $P(\beta+1)$ is true.

By the principle of mathematical induction, we have $\Delta^{v,1}_b \geq \Delta^{v,2}_b, \forall b \in \left\{m, \ldots, n^v\right\}$.■

APPENDIX D: REVISED MODEL WITH VALID INEQUALITIES

In the problem, each station is visited only once by one vehicle only. This implies that each station only needs to do bike pickup or drop-off once. Therefore, we can simplify equations (3)-(6) and (12) to reduce computation time through replacing p_{jv} and r_{jv} by p_j and r_j respectively, in which p_j and r_j represent the number of bikes loaded and unloaded at node *j*. The modified constraints are introduced as follows:

$$\mu \geq \sum_{j \in N} \left(d_j - r_j\right) - TDD, \tag{81}$$

$$p_j - r_j = \sum_{v \in V} \sum_{i \in N_0 \backslash \{j\}} q_{jiv} - \sum_{v \in V} \sum_{i \in N_0 \backslash \{j\}} q_{ijv}, \qquad \forall j \in N; \tag{82}$$

$$p_j \le s_j, \qquad \forall j \in N; \tag{83}$$

$$r_j \le d_j, \qquad \forall j \in N; \tag{84}$$

$$r_j, p_j \ge 0, \text{integer}, \qquad \forall j \in N_0. \tag{85}$$

The expressions of loading and unloading strategies based on Proposition 1 (i.e., equations (17) and (18)) can be re-expressed as

$$p_j = \min\left(s_j, Q - \sum_{v \in V} \sum_{i \in N_0 \setminus \{j\}} q_{ijv} \right), \qquad \forall j \in P; \tag{86}$$

$$r_j = \min\left(d_j, \sum_{v \in V} \sum_{i \in N_0 \setminus \{j\}} q_{ijv} \right), \qquad \forall j \in R. \tag{87}$$

The service time equation (20) can also be modified as

$$S_v = \sum_{i \in N_0} \sum_{j \in N_0 \setminus \{i\}} x_{ijv} \left(p_j L + r_j U \right), \qquad \forall v \in V. \tag{88}$$

Substituting constraint (88) and (9) into objective function (19), the objective function can be rewritten as

$$\min Z = \mu \times M + \sum_{v \in V} T_v + \sum_{j \in N_0} \left(p_j L + r_j U \right). \tag{89}$$

In addition, the following valid inequalities are included in Model 1 to reduce running time:

$$p_i \le \min\left(s_i, Q\right) \sum_{j \in N_0} x_{ijv}, \qquad \forall i \in N, v \in V; \tag{90}$$

$$r_i \le \min\left(d_i, Q\right) \sum_{j \in N_0} x_{ijv}, \qquad \forall i \in N, v \in V; \tag{91}$$

$$\sum_{j \in N} j \cdot x_{0jv} \le \sum_{j \in N} j \cdot x_{0,j,v+1}. \qquad \forall v \in V. \tag{92}$$

After adding these valid inequalities, the resultant mathematical model solved by the CPLEX is formed by objective function (89) and constraints (2), (7)-(11), (13)-(16), (21), (81)-(87), and (90)-(92).