Repositioning Bikes with Carrier Vehicles and Bike Trailers in Bike Sharing Systems

Xinghua Zheng\textsuperscript{1}, Ming Tang\textsuperscript{1}, Hankz Hankui Zhuo\textsuperscript{1,}\textsuperscript{*}, Kevin X. Wen\textsuperscript{2}
\textsuperscript{1}School of Data and Computer Science, Sun Yat-Sen University, Guangzhou, China 510006
\textsuperscript{2}Department of Computer Science, University of California, Santa Cruz, California, US
\{zhengxh5@mail3, tangm28@mail2, zhuohank@mail\}.sysu.edu.cn; kevinxwen@gmail.com

Abstract

Bike Sharing Systems (BSSs) have been adopted in many major cities of the world due to traffic congestion and carbon emissions. Although there have been approaches to exploiting either bike trailers via crowdsourcing or carrier vehicles to reposition bikes in the “right” stations in the “right” time, they do not jointly consider the usage of both bike trailers and carrier vehicles. In this paper, we aim to take advantage of both bike trailers and carrier vehicles to reduce the loss of demand with regard to the crowdsourcing of bike trailers and the fuel cost of carrier vehicles. In the experiment, we exhibit that our approach outperforms baselines in several datasets from bike sharing companies.

Introduction

Bike sharing systems (BSSs) typically have a set of base stations that are strategically placed throughout a city and each station has a fixed number of docks, e.g., Capital Bikeshare\textsuperscript{3}, Bluebikes\textsuperscript{4}, Mobike\textsuperscript{5}, BIXI\textsuperscript{6}, etc. At the beginning of the day, each station is stocked with a pre-determined number of bikes. Customers can pick and drop bikes from any station and are charged depending on the hiring duration (Tsai, Chen, and Hong 2019; Hulot, Aloise, and Jena 2018; Lowalekar et al. 2017; Vulcano, van Ryzin, and Ratliff 2012; Schuijbroek, Hampshire, and van Hoeve 2017).

Due to the individualistic and uncoordinated movements of customers, there is often starvation (empty base stations precluding bike pickup) or congestion (full base stations precluding bike return) of bikes at certain stations, which results in a significant loss of customer demand (Shu et al. 2013; Chen, Liu, and Liu 2018). To address this problem, a variety of systems (Ghosh et al. 2017; Lowalekar et al. 2017) employ the idea of repositioning idle bikes with the help of carrier vehicles during the day, by taking into account the movement of bikes by customers (Tsai, Chen, and Hong 2019; Pfrommer et al. 2014; Ghosh and Varakantham 2017). While previous approaches of repositioning can help reduce imbalance, repositioning idle bikes using carrier vehicles (c.f. (Ghosh, Trick, and Varakantham 2016)) incurs substantial routing and fuel costs while covering entire stations\textsuperscript{7}. In addition, repositioning idle bikes using bike trailers just carries a few of bikes once and the moving distance is limited\textsuperscript{8}, which restrict the usage of bike trailers to reposition bikes among stations.

In this paper, we propose an optimization model called (DRRPVT), which stands for Dynamically Repositioning and Routing Problem with carrier Vehicles and bike Trailers, to jointly consider the usage of carrier vehicles and bike trailers. We aim to better optimize the overall profit of hired bikes and consequently reduce the expected loss of demand. Specifically, we build a profit objective function to calculate the value of carrier vehicle routing (i.e., fuel cost) and bike trailers (i.e., payment for the users of bike trailers), by considering a variety of constraints with respect to carrier vehicle routing and bike repositioning. Jointly considering both carrier vehicles and bike trailers is challenging in the sense that we need to introduce new constraints to encode relations between carrier vehicles and bike trailers, and build a novel objective function to minimize the cost of repositioning (and routing) and the loss of demand. Besides, to improve the efficiency of our approach with respect to large-scale stations (as well as carrier vehicles and bike trailers), we need to design an effective mechanism for computing main base stations to help reduce the computation time.

In summary, our contributions are two folds. We first propose an optimization model to improve the performance of dynamic bike repositioning by exploiting both carrier vehicles and bike trailers simultaneously, which is different from previous approaches which only consider either trailers or carrier vehicles, but not both. To do this, we build a novel profit objective function and new constraints considering relationships between carrier vehicles and bike trailers. Sec-

\textsuperscript{7}A carrier vehicle is a truck to reposition idle bikes during the day using myopic and adhoc methods so as to return to a pre-determined configuration(e.g., each carrier vehicle can hold 30-40 bikes, its working distance is 5 kilometers away).

\textsuperscript{8}A bike trailer is an add-on to a bike that can carry a small number of bikes (e.g., each bike trailer can hold 3-5 bikes, its working distance is within 5 kilometers) and is useful to relocate bikes to nearby stations.
ond, we design a clustering mechanism for computing main base stations to help improve the efficiency of solving the optimization model regarding large-scale stations and carrier vehicles and bike trailers.

Related Work

There have been many approaches proposed to deal with bike sharing issues, which can be categorized into three aspects (Lin, Yang, and Chang 2013, Lowalekar et al. 2017), i.e., static repositioning using carrier vehicles, dynamic repositioning using carrier vehicles, and dynamic repositioning using bike trailers.

**Static repositioning using carrier vehicles** Static repositioning is the problem of finding routes for a fleet of vehicles to reposition bikes at the end of the day when the movements of bikes by customers are negligible, to achieve a predetermined inventory level at the stations (Chemla, Meunier, and Calvo 2013). As user demands change frequently during the day, those approaches are not capable of dynamically adjusting the station inventory level with respect to user demands.

**Dynamic repositioning using carrier vehicles** To consider dynamic repositioning using carrier vehicles with respect to the movements of customers during the day, Lowalekar et al. provide a scalable online repositioning solution using multistage stochastic optimization with online anticipatory algorithms (Lowalekar et al. 2017, Wang et al. 2018). Pierre et al. develop an efficient mechanism to maximize the decision intervals between repositioning events by online rebalancing operations (Hulot, Aloise, and Jena 2018, Chen, Liu, and Liu 2018). As dynamic repositioning using vehicles alone incurs substantial routing and fuel cost, those approaches should be improved by considering self-sustaining and environment friendly.

**Dynamic repositioning using bike trailers** To consider the self-sustaining and environment issues, instead of using vehicles, Ghosh et al. propose a pricing mechanism that takes the global view of the repositioning requirements and incentives the execution of bike-trailer tasks (based on crowdsourcing) within the budget constraints (Ghosh and Varakantham 2017, Singla et al. 2015). Despite the success of those approaches, bike trailers can only take a few bikes at once and the distance of movements is limited. Besides, the value of crowdsourcing tasks may be high (over the available budget).

Different from previous approaches, our DRRPVT approach aims to leverage the advantage of using both carrier vehicles, which is able to take a large number of bikes and move to longer distance, and bike trailers, which is able to move to short distance with limited cost and allow self-sustaining, by considering the expected profit and the loss demands reduction of repositioning and routing solution (Hartuv, Agmon, and Kraus 2018, Zhang and Pavone 2014).

**Problem Formulation**

Our bike sharing problem is formally defined by the following tuple: \((S, V, F, C^#, C^\ast, d^\ast, d^\ast, \sigma, R, P, \hat{P}, D, B)\),

where

- \(S\) denotes the set of base stations.
- \(V\) denotes the set of vehicles used for repositioning which restricted to carrier vehicles only.
- \(F\) denotes samples of customer requests for the future time steps with \(F_{s,s'}^t\) indicating the number of customer requests between stations \(s\) and \(s'\) which start at decision epoch \(t\) and end at decision epoch \(t+1\).
- \(C^\ast\) denotes the capacity of stations with \(C^\ast_s\) indicating capacity of station \(s\).
- \(C^\ast\) denotes the capacity of carrier vehicles with \(C^\ast_v\) indicating capacity of vehicle \(v\).
- \(d^\ast\) denotes the distribution of bikes at stations with \(d^\ast_{s,t}\) indicating the number of bikes at station \(s\) at decision epoch \(t\).
- \(d^\ast\) denotes the distribution of bikes in vehicles with \(d^\ast_v\) indicating the number of bikes in vehicle \(v\) at decision epoch \(t\).
- \(\sigma\) denotes the distribution of carrier vehicles at stations, with \(\sigma^t_{v,s}\) set to be 1 if vehicle \(v\) is present at station \(s\) at decision epoch \(t\) and 0 otherwise.
- \(R\) denotes the revenue of bikes being hired, with \(R^t_{s,s'}\) indicating the revenue from station \(s\) to \(s'\) which starts at decision epoch \(t\) and ends at decision epoch \(t+1\).
- \(D\) denotes the actual distance with \(D_{s,s'}\) indicating the distance between stations \(s\) and \(s'\).
- \(B\) denotes the total budget for all trailers to bid. In other words, the total amount of value spent on trailers should not be larger than \(B\).
- \(\hat{P}\) denotes the value for executing the task of bike trailer with \(\hat{P}_{s,s'}\) indicating the value for executing the task of bike trailer picking up idle bikes at station \(s\) and dropping them at station \(s'\).
- \(P\) denotes the routing value (e.g., fuel cost) for vehicles travelling with \(P_{s,s'}\) indicating the routing value for vehicles travelling from station \(s\) to \(s'\) which depends on the distance between the two stations.

We make the following assumptions for the ease of explanation and representation:

1. We assume that users who carry bikes and trailers at decision epoch \(t\) always return their bikes at the beginning of the decision epoch \(t+1\). The duration of each decision epoch is 30 minutes\(^7\).

2. We sampled the empirical distribution of the real historical data of customer requests to simulate customer requests for the future time steps (Frommer et al. 2014). We assume that the lost demand at the time of return. Once the distribution of bikes across the stations for time step \(t+1\) is obtained, we utilize this information to compute the repositioning strategy for trailers and vehicles for time

\(^7\)We evaluate shorter duration impacts on runtime performance. Reducing the duration of time step notably increases the runtime. There is a trade-off between utility and runtime in deciding the duration of time step. Although the performance in terms of profit and lost demand decreases by a small amount for 30 minutes of time step (over 15 minutes of time step), it provides a significant computational gain and is particularly helpful when solving large problems. Therefore, we choose 30 minute as the default setting for the duration of time step.
step \( t + 1 \). This iterative process continues until we reach the last decision epoch;

3. Customers can rent a bike for 30 minutes or more, and they have to know in advance at which station they will return the bike. On the other hand, they return their bikes to the nearest available station if the destination station is full, and they leave the system if they encounter an empty station.

The goal of our DRRPVT approach is to maximize the expected profit over the entire time horizon. Let \( U \) denotes the sum of revenue of hired bikes and the fuel cost of vehicles and the value of bike trailers. We provide an optimisation model for a given DRRPVT. Specifically, we provide a mixed integer linear programming (MILP for short) that computes a profit maximising repositioning and routing solution. The objective is shown in Equation (1):

\[
\max_{y,z,a,b} U = \max_{y,z,a,b} \sum_{s,s',t} R_{s,s'}^t \times x_{s,s'}^t - \sum_{t,v,s,s'} P_{s,s'} \times a_{s,s',v}^t - \sum_{t,v,s,s'} \hat{P}_{s,s'} \times b_{s,s',v}^t
\]

s.t. constraints C1-C15 which depends on \( y, z, a, b \).

**Objective:** To represent the trade-off between lost demand (or alternatively the revenue from customer trips) and the value of bike trailers. We provide an optimisation model for a given DRRPVT. Specifically, we provide a mixed integer linear programming (MILP for short) that computes a profit maximising repositioning and routing solution. The objective is shown in Equation (1):

\[
\max_{y,z,a,b} U = \max_{y,z,a,b} \sum_{s,s',t} R_{s,s'}^t \times x_{s,s'}^t - \sum_{t,v,s,s'} P_{s,s'} \times a_{s,s',v}^t - \sum_{t,v,s,s'} \hat{P}_{s,s'} \times b_{s,s',v}^t
\]

s.t. constraints C1-C15 which depends on \( y, z, a, b \).

C1: Preservation of Bike Flows in and out of station. We require that the number of bikes in a vehicle at a time step \( (d^t_v) \) is equivalent to the sum of the number of bikes in the vehicle at the previous time step \( (d^t_{v-1}) \) and the net number of bikes coming into the vehicle during that time step, i.e., for each station \( s \) and epoch \( t \),

\[
d^t_v + \sum_s (y_s^t - y_s^{t-1}) = d^t_v + \sum_s (y_s^t - y_s^{t-1} + a_{s,v}^t - a_{s,v}^{t-1})
\]

where the net number is defined by the last three components.

C2: Preservation of Bikes Flows between any two stations follow the transition dynamics observed in the data. As a subset of arrival demand can be served if the number of bikes present in a station is less than the arrival demand, we require that bikes flows between station \( s \) and \( s' \) should be less than the product of the number of bikes present in the source station \( s \) \((d^t_{s,v})\) and the transition probability that a bike will move from \( s \) to \( s' \) according to expected customer demand, i.e., for each \( t, s, s', x_{s,s'}^t \leq d_{s,v}^t \times \sum_{s,s,v} P_{s,s'} \times x_{s,s'}^t \).

C3: Value of task for bike trailer. We require a mechanism for crowdsourcing the repositioning tasks to the users of bike trailers and generating a payment method to ensure that the users bid for the tasks truthfully. The valuation of trailer \( v \) task is proportional to the expected lost demand reduced by the trailer job in the training demand scenario. \( \xi \) represents unit value of lost demand to compute overall value, i.e., \( \sum_{s,s',v} \hat{P}_{s,s'} \times \sum_{s,s,v} P_{s,s'} \times x_{s,s'}^t \leq B \).

C4: Ensuring the Budget Feasibility. We require to incentive compatibility over all tasks without violating the fix budget \( B \) feasibility. Each task of trailers \( v \in V \) has a valuation for the task is denoted by \( \hat{P} \). We aim to allocate the tasks in a fashion that maximizes the overall valuation of the center while the total payment is bounded by the given budget \( B \), i.e., \( \sum_{s,s',v} b_{s,s',v}^t \times \hat{P}_{s,s'} \leq B \).

C5: Preservation of Bikes Flows in and out of vehicles. We require that the number of bikes in a vehicle at a time step \( (d^t_{s,v}) \) is equivalent to the sum of the number of bikes in the vehicle at the previous time step \( (d^t_{s,v-1}) \) and the net number of bikes coming into the vehicle during that time step \( (\sum_s (y_s^t - y_s^{t-1}) \text{ i.e., for each } v, t, d_{s,v}^t + \sum_s (y_s^t - y_s^{t-1}) = d_{s,v}^t \times \sum_{s,s,v} P_{s,s'} \times x_{s,s'}^t \).

C6: Preservation of Vehicles Flows in and out of stations. We require that the number of vehicles going out of station \( s \) \((\sum_s z_{s,v}^{t-1}) \) plus the number of vehicles present at station \( s \) at time epoch \( t-1 \) \((\sigma_{s,v}^{t-1}) \) is equivalent to the sum of the number of vehicles coming into station \( s \) \((\sum_s z_{s,v}^t) \) and the vehicles which are present at station \( s \) at time epoch \( t \) \((\sigma_{s,v}^t) \).

C7: A maximum of one vehicle can be present in one station at any time step. Due to limited space availability near base stations and to avoid a synchronisation issue in pickup or drop-off events by multiple vehicles from the same station at the same time step, we require that the maximum number of vehicles at a station \((\sum_{s,v} z_{s,s,v}^t) \) is less than 1, i.e., for each \( t, s, \sum_{s,v} z_{s,s,v}^t \leq 1 \).

**Constraints**

In this section, we address the constraints (C1-C15) we exploit in our bike sharing system, where constraints (C1-C4) are newly created in this paper, while constraints (C5-C8) have been presented by Lowalekar et al. (2017) and Ghosh et al. (2017) and constraints (C9-C15) have been presented by Varakantham (2017).
C8: Vehicles can only pick up or drop off bikes at a station if they are present at that station. We require that the number of bikes picked up or dropped off at station at each time step by each vehicle is bounded by whether the station is visited by the vehicle at that time step or not, i.e., for each $s, v, t$, $y_{s,v}^+, t + y_{s,v}^- t \leq C^*_v \times \sum_{s,s',v} z^t_{s, s', v}$.  

C9: Trailer capacity is not exceeded while picking up bikes. We require that the number of bikes picked up by trailer $v$ from station $s$ is bounded by the minimum value between the number of bikes present in the station and the capacity of the trailer. $b_{s,s',v}^t$ denotes a binary decision variable which is set to 1 if bike trailer $v$ picks up bikes from station $s$ and drop off bikes to any station $s'$ and 0 otherwise, i.e., for each $s, v, t, a_{s,v}^t \leq \sum_{s,s'} b_{s,s',v}^t \times \min(a_{s,v}^t, C^*_v)$.  

C10: Total number of bikes picked up from a station is less than the available bikes. As multiple trailers can pick up bikes from the same station, we require that the total number of picked up bikes by all the trailers from station $s$ during the planning period $t$ is bounded by the number of bikes present at the station ($d_{s,v}^t$), i.e., for each $s, v, t$, $\sum_{s,s'} a_{s,v}^t \leq d_{s,v}^t$.  

C11: Station capacity is not exceeded while dropping off bikes. We require that the total number of dropped off bikes at station $s$ is bounded by the number of available slots for bikes at that station, i.e., for each $s, v, t$, $\sum_{s,s'} a_{s,v}^t \leq C^*_v - d_{s,v}^t$.  

C12: Total travelling distance for a trailer is bounded by a threshold value. To represent the physical limitation of route, we need to ensure that the total distance travelled by a trailer in a given planning period is within a few kilometers. We require that the distance between pick-up station and the drop-off station for a trailer $v$ is bounded by a threshold value, $D_{\text{max}}$, i.e., for each $s, s', v, t, b_{s,s',v}^t \times D_{s,s'} \leq D_{\text{max}}$.  

C13: A trailer can only pick up or drop off bikes at exactly one station. We require that a trailer can go to exactly one station starting from a specific station, i.e., for each $v, t$, $\sum_{s,s'} b_{s,s',v}^t = 1$.  

C14: A trailer should return the exact number of bikes picked up. We require that the number of bikes dropped off by a bike trailer in a station is exactly equal to the number of bikes picked up if the station is visited, i.e., for each $s', v, t$, $a_{s',v}^- t = \sum_{s,s'} (b_{s,s',v}^t \times a_{s,v}^t)$. Note that, above equation are non-linear in nature. However, one component in the right hand side is a binary variable. Therefore, we can easily linearize them using the following formula, i.e., for each $s', v, t, a_{s',v}^- t \leq C^*_v \times \sum_{s,s'} b_{s,s',v}^t \times a_{s,v}^t \leq \sum_{s,s'} a_{s,v}^t, a_{s',v}^- t \geq \sum_{s,s'} a_{s,v}^t - (1 - \sum_{s,s'} b_{s,s',v}^t) \times C^*_v$.  

C15: Station and vehicle capacities are not exceeded when repositioning bikes. We require that the number of bikes at a station $s$ does not exceed the number of available docks at that station ($C^*_s$). Similarly, these constraints also enforce that the number of bikes picked up or dropped off by a vehicle $v$ in aggregate does not exceed the capacity of the vehicle ($C^*_v$), i.e., $0 \leq x_{s,v}^t \leq F^t_{s,v}; 0 \leq d^t_{s,v} \leq C^*_v; 0 \leq d^t_{v} \leq C^*_v; 0 \leq y_{s,v}^+, t + y_{s,v}^- t \leq d_{s,v}^t; 0 \leq y_{s,v}^-, t + a_{s,v}^- t \leq d_{s,v}^t; 0 \leq x_{s,v}^+, t + x_{s,v}^- t \leq C^*_v; 0 \leq y_{s,v}^+, t + y_{s,v}^- t \leq d_{s,v}^t; 0 \leq x_{s,v}^+, t + x_{s,v}^- t \leq C^*_v; 0 \leq y_{s,v}^-, t + a_{s,v}^- t \leq d_{s,v}^t; 0 \leq y_{s,v}^+, t + y_{s,v}^- t \leq d_{s,v}^t; 0 \leq a_{s,v}^- t \leq b_{s,s',v}^t \times (0, 1)$.  

Given C1-C15, our task is to calculate which vehicles reposition bikes from state $s$ to $s'$, i.e., $z$, and which trailers reposition bikes from $s$ to $s'$, i.e., $b$, by optimizing Equation (1).  

Our DRRPVT Approach  

In order to solve Equation (1), we use the well-known Lagrangian dual decomposition (LDD) (Fisher, 1985; Gordon et al., 2012) technique. While this is a general purpose approach, its scalability, usability and utility depend significantly on the following characteristics of the model:  

Identifying the right constraints to be dualized: This step is crucial to ensure that the resulting subproblems are easy to solve and the resulting bound derived from the dual solution is tight during the LDD process. If the right constraints are not dualized, then the underlying Lagrangian based optimization may not be decomposable or it may take significantly more time than the original MILP to find the desired solution.  

Extraction of a primal solution from an infeasible dual solution: The primal extraction process is important to derive a valid bound (heuristic solution) during the LDD process. In many cases, the solution obtained by solving the decomposed dual slaves can be infeasible with respect to the original formulation and hence, the overall approach can potentially lead to slower convergence and poor solutions.  

Decompose the original problem into a master problem and two slaves (SOLVERDEPLOY and SOLVEROUTING): As highlighted in Equation (1), only constraints (8) contain a dependency between routing and repositioning variables. We dualize constraints (8) using the dual variables, $\alpha_{s,v,t}$ and obtain the Lagrangian function as Equation (2).  

We exploit LDD to provide a near optimal solution for the dynamic repositioning of bikes (Ghosh et al., 2017; Ghosh et al., 2013). Although the LDD framework was indeed used in Ghosh et al., 2015 and 2017, challenging to investigate the usage of LDD to accommodate the new constraints. An overview of DRRPVT is shown in Algorithm 1.  

We will present main steps of Algorithm 1 in the following subsections.  

Algorithm 1 An overview of our DRRPVT approach  

**Input:** $(S, V, F, C^#, C^*, d^#, d^*, \sigma, P, \hat{P}, R, D, B)$  

**Output:** $y, z, a, b$  

1: $S \leftarrow \text{CalculateMainStations}(S, D)$  
2: $\alpha = 0$, $iter = 0$  
3: while $[p - (\rho_1 + \rho_2)] \leq \delta$ do  
4: $\rho_1, y, a, b \leftarrow \text{SolveReposition}(\alpha_{iter})$  
5: $\rho_2, z \leftarrow \text{SolveRouting}(\alpha_{iter})$  
6: $\alpha_{iter+1} \leftarrow [\alpha_{iter} + \gamma \times [(y^+ + y^- - C^* \times \sum z_{s,v})] +$  
7: $p, y, a, b \leftarrow \text{ExtractPrimal}(z)$  
8: $iter \leftarrow iter + 1$  
9: end while  
10: $\text{SolvingIncentivizeTrailerTask}(d^#, F, a)$  

Our task is to optimize Equation (1) to calculate $y, a, b, z$. 

To do this, based on Equation (1), we can define a Lagrangian function as shown below:

\[
L(\alpha) = \min_{y,a,b} \left[ -\sum_{t,s,s'} R_{t,s,s'}^t \times x_{t,s,s'}^t + \sum_{t,s,v,t} z_{t,s,s',v} \times P_{s,s'} + \sum_{s,s',v,t} b_{s,s',v}^t \times \tilde{P}_{s,s'} + \sum_{s,v,t} \alpha_{s,t,v} \left( y_{s,v}^t + \tilde{y}_{s,v}^t \right) + \sum_{s,s',v,t} b_{s,s',v}^t + \min_{z} \sum_{s,s',v,t} z_{s,s',v} \times \left( P_{s,s'} \times C_v^* \times \alpha_{s,t,v} \right) \right]
\]

\[
s.t. \quad \text{Constraints C1} - \text{C7} \quad \text{and C9} - \text{C15} \quad (2)
\]

, which is equivalent to

\[
L(\alpha) = \min_{y,a,b} \left[ -\sum_{t,k,s,s'} R_{t,s,s'}^t \times x_{t,s,s'}^t + \sum_{s,v,t} \alpha_{s,t,v} \left( y_{s,v}^t + \tilde{y}_{s,v}^t \right) + \sum_{s,s',v,t} b_{s,s',v}^t + \min_{z} \sum_{s,s',v,t} z_{s,s',v} \times \left( P_{s,s'} \times C_v^* \times \alpha_{s,t,v} \right) \right]
\]

\[
s.t. \quad \text{Constraints C1} - \text{C7} \quad \text{and C9} - \text{C15} \quad (3)
\]

Calculating Main Stations

Since nearby stations can be covered by bike trailers, we exploit the geographical proximity based clustering method to obtain main stations to reduce the usage of carrier vehicles (Ghosh et al. 2017; Gaspero, Rendl, and Urli 2016). We thus provide a clustering mechanism to calculate main stations in Step 1 of Algorithm 1. The high-level idea is to first calculate distances between base stations, and then cluster base stations based on their distances using off-the-shelf clustering approaches such as k-means. We denote the set of resulting main stations by \( \tilde{S} \) (Ghosh and Varakantham 2018; Konda, Ghosh, and Varakantham 2018; Jha et al. 2018). Therefore, we utilize carrier vehicles to reposition bikes dynamically for a small range (i.e., within each main station).

Repositioning Bikes and Routing for Vehicles

Our goal is to design a mechanism to incentivize task execution based on the maximization of profit via dynamically repositioning and routing. Specifically, we provide a decomposition approach to exploit the minimal dependency that exists in the model DRRPV in conjunction with the repositioning problem (how many bikes to pick up and drop off at each station) and the routing problem (how to move vehicles between base stations to pick up or drop off bikes). The following observation highlights this minimal dependency:

- \( y, a, b \) capture the solution to the repositioning problem.
- \( z \) captures the solution to the routing problem.

These sets of variables only interact with each other in constraint (8). In all of the other constraints of our DRRPV model, the routing variables are completely independent with repositioning variables.

With the minimal dependency observation, we use LDD in DRRPV. It is crucial to ensure that the resulting subproblems are easy to solve and the resulting bound derived from the dual solution is tight during the LDD process. We first decompose the original problem into a master problem (i.e., Equation (3)) and two slaves SolveReposition and SolveRouting. As highlighted, only constraint (8) contains dependencies between routing and repositioning variables, i.e., \( \alpha_{s,t,v} \). Thus, we dualize constraint (8) using the dual variables, and obtain the Lagrangian function in Equation (3). The first three terms in Equation (3) corresponding to the repositioning problem are given in Equation (4), and the last term corresponding to the routing problem is given in Equation (5), respectively, i.e.,

\[
\min_{y,a,b} \sum_{s,v,t} \alpha_{s,t,v} \times y_{s,v}^t + \sum_{s,s',v,t} b_{s,s',v}^t \times \tilde{P}_{s,s'} + \sum_{s,s',v,t} R_{s,s',v}^t \times x_{s,s',v}^t
\]

\[
s.t. \quad \text{Constraints C1} - \text{C5} \quad \text{and C9} - \text{C15} \quad (4)
\]

and

\[
\min_{z} \sum_{s,s',v,t} z_{s,s',v} \times \left( P_{s,s'} - C_v^* \times \alpha_{s,t,v} \right)
\]

\[
s.t. \quad \text{Constraints C6-C7} \quad \text{and C15} \quad (5)
\]

From Equation (3), given \( \alpha \), the dual value corresponding to the original problem is obtained by adding up the objective function values from the two slaves, which yields a valid lower bound with respect to the original problem. It should be noted that the decomposition is only for \( L(\alpha) \). The value of \( \text{SolveReposition} \) is denoted by \( \rho_1 \), and The value of \( \text{SolveRouting} \) is denoted by \( \rho_2 \).

Next, we solve the following optimization problem at the master in order to reduce violations of the dualized constraints:

\[
\max_{\alpha \geq 0} L(\alpha).
\]

This master optimization problem is solved iteratively using a sub-gradient descent method applied on the dual variables \( \alpha \), i.e., Step 6 of Algorithm 1, where \( \gamma \) is a step-size parameter. The algorithm terminates when the difference between the primal objective (defined as \( p \) in Algorithm 1) and the dual objective (the sum of the slaves objectives \( \rho_1, \rho_2 \)) is less than a pre-determined threshold value \( \delta \). In order to compute the best primal solution in conjunction with the dual solution, it is important to obtain a primal solution after each iteration from the solutions of the slaves. The infeasibility in the dual solution arises because the routes of the vehicles (obtained by solving the routing slave) may not be consistent with the repositioning plan of bikes (obtained by solving the repositioning slave). However, the solution for the routing slave is always feasible and can be fixed to obtain a feasible primal solution with respect to the original problem. Let \( z_{s,s',v}^* = \sum_{s'} z_{s,s',v}^* \). We extract the primal solution by solving the optimization formulation in Equation (6):

\[
\max_{y} \sum_{t,s,s'} R_{t,s,s'}^t \times x_{t,s,s'}^t - \sum_{t,v,s,s'} b_{t,s,v}^t \times \tilde{P}_{s,s'}
\]

\[
s.t. \quad \text{Constraints C1} - \text{C5} \quad \text{and C9} - \text{C15} \quad (6)
\]

\[
y_{s,v}^+,t + y_{s,v}^-,t \leq C_v^* \times z_{s,v}^* \quad \forall t, s, v
\]
Specifically, constraints in Equation (6) are equivalent to constraint (8) where we use the solution values of the routing slave \( z \) as the input. Thus, ExtractPrimal satisfies C1-C5,C9-15 and produces a feasible solution to the original problem. Finally, we subtract the routing value from the objective value to get the correct primal value.

**Incentivize Trailer Tasks**

In Step 10, we use an incentivizing mechanism proposed by Ghosh and Varakantham (2017), which allocates the tasks to users of bike trailers. Firstly, the mechanism computes the value of the tasks according to the lost demand reduced by the trailer task. Secondly, it employs an incentive compatible mechanism that ensures users always bid truthfully on each task. Finally, it assigns the task to a bidder so that the profit is maximized, and employs a payment method to ensure that the task is always allocated to the lower bidder. The total payment given to the users of trailers due to the resulting allocation should respect to the given budget \( B \).

**Experiments**

To exhibit the effectiveness of our approach, we conducted the experiment on two datasets Capital Bikeshare\(^8\) and Hubway\(^9\) and a synthetic dataset which was derived from multiple real datasets. We generated the synthetic dataset by first taking a subset of the stations from the two real-world datasets, and then taking customer demand, station capacity, geographical location of stations, initial distribution, bid values and value model drawn from the two real-world datasets. The Hubway dataset consists of 95 base stations and 3 vehicles, 10 trailers; Capital Bikeshare dataset consists of 305 active stations and 10 vehicles, 35 trailers; and the synthetic dataset consists of 60 base stations, 2 vehicles, 7 trailers. We employed k-means clustering to generate 12 main stations (5 base stations are grouped into 1 main station) and main stations in (Lowalekar et al. 2017), respectively. We then compare LDD and main stations in DRRPV with MILP to see the advantage of LDD and main stations. We finally evaluate DRRPVT remains robust with respect to variation of the numbers of stations, vehicles and trailers.

**Experimental Results**

**Comparison against Baselines** We provide the key performance comparison with respect to the overall profit to show that we can reduce the lost demand without incurring extra value to the operators. We employ 3 vehicles and 20 bike trailers for the experiments in both Capital Bikeshare and Hubway, which is also exploited by Ghosh and Varakantham (2017). We evaluate DRRPVT with respect to different time periods, i.e., the peak period and the whole day. Tables 1 and 2 show the average percentage gain in profit and reduction in lost demand with our approach in comparison to the baselines on the two real-world datasets. Based on the aggregate results, our approach DRRPVT is always able to outperform both DRRPV and DRRPT with respect to both of the profit gain and lost demand reduction. From Table 1, our approach performs much better in Hubway than Capital Bikeshare comparing to baselines. This is because the number of users hiring bikes in Hubway is much larger than Capital Bikeshare. The more users hire bikes, the better our approach performs. Similar results can be found in Table 2.

Lastly, to see the effect of repositioning, we draw the correlation between the actual demand and the served demand over decision epoch. Figure 1 shows the correlation by running the three approaches. Each point in the figure corresponds to the values of an actual demand and its corresponding served demand for all time steps and all stations in the Hubway data set. As expected, our approach has significantly more points closer to the identity line than the other two, which indicates our approach is able to better match the supply of bikes with the demand for bikes.

---

\(^8\)http://www.capitalbikeshare.com/system-data  
\(^9\)http://hubwaydatachallenge.org/trip-history-data

---

Table 1: Comparison of profit gain and lost demand reduction in a whole day (5AM-12AM)

| datasets       | \(\psi_v\) | \(L_v\) | \(\phi_t\) | \(L_t\) |
|----------------|-----------|--------|-----------|--------|
| Hubway         | 2.42%     | 23.57% | 2.18%     | 26.91% |
| Capital Bikeshare | 1.97%   | 14.42% | 1.25%     | 17.38% |

Table 2: Comparison of profit gain and lost demand reduction in the peak period (5AM-12PM)

| datasets       | \(\psi_v\) | \(L_v\) | \(\phi_t\) | \(L_t\) |
|----------------|-----------|--------|-----------|--------|
| Hubway         | 4.63%     | 29.71% | 4.26%     | 31.12% |
| Capital Bikeshare | 4.25%   | 19.39% | 4.11%     | 24.45% |

We would like to verify the following aspects\(^10\) We first evaluate that our DRRPVT approach with novel mechanism (LDD + Main station) outperforms two baselines which use vehicles (Lowalekar et al. 2017) and trailers (Ghosh and Varakantham 2017), respectively. We then compare LDD and main stations in DRRPVT with MILP to see the advantage of LDD and Main stations. We finally evaluate DRRPVT remains robust with respect to variation of the numbers of stations, vehicles and trailers.

---

\(^10\)All optimization models were solved based on GUROBI 7.5.2 and 4.0 GHz Intel Core i7.
Comparison with MILP We next compare LDD and Main stations of DRRPVT to MILP with respect to runtime performance, duality gap and main stations.

Runtime performance: We compare the runtime of DRRPVT with MILP, as shown in Figure (2a). The X-axis denotes the number of stations from 5 to 60. The Y-axis denotes the total time taken to solve problem in seconds. We can see that DRRPVT generally outperforms MILP with respect to number of stations. MILP is unable to finish within a cut-off time of 3 hours for any problem with more than 20 stations, while DRRPVT is able to obtain near optimal solutions on problems with 60 stations in less than 3 hours. DRRPVT becomes relatively stable after reaching 35 stations (the red curve). It could be easily speeded up by running our approach in a server of higher performance in real-world applications. Meanwhile, we observed the trend in runtime when using main station clustering on problems with 100-200 stations and it scaled in similar trend with respect to using v.s. not using main stations.

Duality gap: We demonstrate the convergence of LDD to near optimal solutions. LDD achieves an optimal solution if the duality gap, i.e., the gap between primal and dual solutions, becomes zero. Figure (2b) shows that the duality gap for the instances with 30 stations (grouped into 6 main stations). For these larger problems we are able to obtain a solution with the duality gap of less than 1%.

Main stations: We also would like to demonstrate the performance of the clustering method in comparison with the optimal solution of instances with 30 base stations (grouped into 6 main stations). Table 3 shows the effect of using main stations on the generated profit and runtime based on five random scenarios of customer demand. With main stations, there is obviously an improvement of more than 13% in profit on average over all of the optimal solutions from Table 3. Since main stations are based on geographical proximity, it is ideally suitable for handling such scenarios.

Varying numbers of stations, vehicles and trailers

We compare the profit of DRRPVT with the ratio of base stations over main stations, as shown in Figure (3b). We then compare the profit of DRRPVT with the ratio of base stations over carrier vehicles, as shown in Figure (3b). Finally, we evaluate the profit of the DRRPVT with the ratio of base stations over bike trailers, as shown in Figure (3c).

From Figure 3, we can see that the profit of our DRRPVT approach generally increases at the beginning, with respect to the increase of the ratios of base stations over main stations, carrier vehicles and bike trailers, respectively. After the profit reaches the maximal value, it goes down when the ratios increase. This is consistent with our intuition since more base stations can indeed raise the profit on repositioning bikes at the beginning. It will, however, largely raise the cost of repositioning bikes when base stations become too many.

Conclusion

In this paper we propose an optimization model to jointly consider the usage of carrier vehicles and bike trailers. We build a profit objective to calculate the value of carrier vehicle routing and bike trailers by considering a variety of constraints with respect to vehicle routing and bike repositioning. In the experiment, we exhibit that our approach is effective with comparison to baselines. In the future, it would be interesting to study a budget feasible mechanism which solves the uncertainties in completion time of trailer tasks and build an iterative scenario generation approach which provides the update strategies for pre-planned solutions. In this work, we consider building an
objective function and optimizing the objective according to a set of constraints. The constraints are numerous and sometime difficult to create by hand. It would be interesting to study the feasibility of exploiting classical planning models, such as PDDL (Geffner 2003), with state-of-the-art PDDL model learning approaches (Zhuo et al. 2010, Zhuo, Nguyen, and Kambhampati 2013, Zhuo and Kambhampati 2013, Zhuo and Yang 2014, Zhuo and Kambhampati 2015, Zhuo and Kambhampati 2017) to learn PDDL models from training data automatically, instead of building constraints manually.

References

[Cavallo 2009] Cavallo, R. 2009. Mechanism design for dynamic settings. SIGecom Exchanges 8(2):7.

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

[Chen, Liu, and Liu 2018] Chen, Q.; Liu, M.; and Liu, X. 2018. Bike fleet allocation models for repositioning in bike-sharing systems. IEEE Intel. Transport. Syst. Mag. 10(1):19–29.

[Gaspero, Rendl, and Urli 2016] Gaspero, L. D.; Rendl, A.; and Urli, T. 2016. Balancing bike sharing systems with constraint programming. Constraints 21(2):318–348.

[Geffner 2003] Geffner, H. 2003. PDDL 2.1: Representation vs. computation. J. Artif. Intell. Res. 20:139–144.

[Ghosh and Varakantham 2017] Ghosh, S., and Varakantham, P. 2017. Incentivizing the use of bike trailers for dynamic repositioning in bike sharing systems. In Proceedings of ICAPS, 373–381.

[Ghosh and Varakantham 2018] Ghosh, S., and Varakantham, P. 2018. Dispatch guided allocation optimization for effective emergency response. In Proceedings of AAAI, 775–783.

[Ghosh et al. 2015] Ghosh, S.; Varakantham, P.; Adulyasak, Y.; and Jaillet, P. 2015. Dynamic redeployment to counter congestion or starvation in vehicle sharing systems. In Proceedings of ICAPS, 79–87.

[Ghosh et al. 2017] Ghosh, S.; Varakantham, P.; Adulyasak, Y.; and Jaillet, P. 2017. Dynamic repositioning to reduce lost demand in bike sharing systems. J. Artif. Intell. Res. 58:387–430.

[Ghosh, Trick, and Varakantham 2016] Ghosh, S.; Trick,
M.; and Varakantham, P. 2016. Robust repositioning to counter unpredictable demand in bike sharing systems. In Kambhampati, S., ed., Proceedings of IJCAI, 3096–3102.

[Hartuv, Agmon, and Kraus 2018] Hartuv, E.; Agmon, N.; and Kraus, S. 2018. Scheduling spare drones for persistent task performance under energy constraints. In Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2018, Stockholm, Sweden, July 10-15, 2018, 532–540.

[Hulot, Aloise, and Jena 2018] Hulot, P.; Aloise, D.; and Jena, S. D. 2018. Towards station-level demand prediction for effective rebalancing in bike-sharing systems. In Proceedings of SIGKDD, 378–386.

[Hulot, Aloise, and Jena 2018] Hulot, P.; Aloise, D.; and Jena, S. D. 2018. Towards station-level demand prediction for effective rebalancing in bike-sharing systems. In Proceedings of SIGKDD, 378–386.

[Konda, Ghosh, and Varakantham 2018] Konda, M.; Ghosh, S.; and Varakantham, P. 2018. Reserved optimisation: Handling incident priorities in emergency response systems. In de Weerdt, M.; Koenig, S.; Röger, G.; and Spaan, M. T. J., eds., Proceedings of ICAPS, 330–338. AAAI Press.

[Lin, Yang, and Chang 2013] Lin, J.; Yang, T.; and Chang, Y. 2013. A hub location inventory model for bicycle sharing system design: Formulation and solution. Computers & Industrial Engineering 65(1):77–86.

[Lowalekar et al. 2017] Lowalekar, M.; Varakantham, P.; Ghosh, S.; Jena, S. D.; and Jailed, P. 2017. Online repositioning in bike sharing systems. In Proceedings of ICAPS, 200–208.

[Pfrommer et al. 2014] Pfrommer, J.; Warrington, J.; Schildbach, G.; and Morari, M. 2014. Dynamic vehicle redistribution and online price incentives in shared mobility systems. IEEE Trans. Intelligent Transportation Systems 15(4):1567–1578.

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

[Shu et al. 2013] Shu, J.; Chou, M. C.; Liu, Q.; Teo, C.; and Wang, I. 2013. Models for effective deployment and redistribution of bicycles within public bicycle-sharing systems. Operations Research 61(6):1346–1359.

[Singla et al. 2015] Singla, A.; Santoni, M.; Bartók, G.; Mukerji, P.; Meenen, M.; and Krause, A. 2015. Incentivizing users for balancing bike sharing systems. In Proceedings of AAAI, 723–729.

[Tsai, Chen, and Hong 2019] Tsai, M.; Chen, P.; and Hong, Y. J. 2019. Enhancing the utilization of public bike sharing systems using return anxiety information. Future Generation Comp. Syst. 92:961–971.

[Vulcano, van Ryzin, and Ratliff 2012] Vulcano, G. J.; van Ryzin, G. J.; and Ratliff, R. 2012. Estimating primary de-