Efficient Spatial-Temporal Rebalancing of Shareable Bikes (Student Abstract)

Zichao Deng,1,2 Anqi Tu,1 Zelei Liu,1 Han Yu1
1School of Computer Science and Engineering, Nanyang Technological University (NTU), Singapore
2Alibaba-NTU Singapore Joint Research Institute
{zichao001, atu001}@e.ntu.edu.sg, {zelei.liu, han.yu}@ntu.edu.sg

Abstract
Bike sharing systems are popular worldwide now. However, these systems are facing a problem - rebalancing of shareable bikes among different docking stations. To address this challenge, we propose an approach for the spatial-temporal rebalancing of shareable bikes which allows domain experts to optimize the rebalancing operation with their knowledge and preferences without relying on learning by trial-and-error.

Introduction
Bike sharing systems provide short-distance bike rental services for commuters with many docking stations scattering over an urban area. These systems are used to bridge the gap between existing transportation modes and promote the efficiency of commuting in a sustainable way. However, due to differences in road congestion and budget limitations, shared parking stations face problems - rebalancing bikes at time slot t (users dropping off bikes); μi(t) is the number of outgoing bikes from docking station i at time slot t (users picking up bikes); and ai(t) is the number of bikes which should be moved into or out of docking station i at time slot t by trucks as part of the bike rebalancing operation (i.e., the control variable in our problem).

We design the following Lyapunov function to model the distribution of bikes among docking stations: L(t) = \frac{1}{2} \sum_{i=1}^{N} (q_i(t) - \delta_i(t))^2, where \delta_i(t) \geq 0 denotes the target number of bikes at docking station i at time slot t in anticipation of demand in the near future. The time-averaged conditional Lyapunov drift is expressed as: Δ = \frac{1}{T} \sum_{t=0}^{T-1} \{L(t+1) - L(t)|q(t)}\). If normal usage results in large fluctuations in Δ, rebalancing interventions need to be carried out in order to improve the operational efficiency.

The time-averaged cost for moving bikes between docking stations by trucks can be approximated as: C = \frac{1}{T} \sum_{t=0}^{T-1} \sum_{i=1}^{N} a_i(t)\delta_i(t). At different times of a day, the cost of moving bikes around by trucks may differ. This is mainly due to differences in road congestion levels. Overall, we aim to minimize a joint objective function of \{cost+drift\} which can be expressed as φC + Δ, where φ > 0 is a weight factor a system operator can use to express his preference of cost saving over balancing demand and supply among bike docking stations. Therefore, the approach will minimize f(t) = \frac{1}{T} \sum_{t=0}^{T-1} \sum_{i=1}^{N} \left\{ \frac{1}{2} a_i^2(t) + a_i(t)\mu_i(t) + a_i(t)\lambda_i(t) - \delta_i(t) \right\}, subject to \sum_{t=0}^{T-1} a_i(t)\delta_i(t) \leq B(t), \forall a_i(t) > 0, \forall t where B(t) is the total budget available for rebalancing bikes at time slot t.

The Proposed STR Approach
A given shareable bike docking station i can be modelled as a queueing system. The dynamics of qi(t) is as follows: qi(t + 1) = \max\{qi(t) + \lambda_i(t) + a_i(t) - \mu_i(t), 0\}, where qi(t) is the number of bikes at docking station i at time slot t; \lambda_i(t) is the number of incoming bikes to docking station i at time slot t (users dropping off bikes); \mu_i(t) is the number of outgoing bikes from docking station i at time slot t (users picking up bikes); and ai(t) is the number of bikes which should be moved into or out of docking station i at time slot t by trucks as part of the bike rebalancing operation (i.e., the control variable in our problem).
In this paper, we do not focus on computing \( \delta_t(t+1) \), but delegate to existing short-term shareable bike demand forecasting models such as (Pan et al. 2019). We approximate \( a_t(t) \) with the variable \( \hat{a}_t(t) \) through a floor operator to ensure it is an integer: 
\[
\hat{a}_t(t) = \lfloor a_t(t) \rfloor = \lfloor \delta_t(t+1) + \mu_t(t) - q_t(t) - \lambda_t(t) - \rho c(t) \rfloor.
\]
During a time slot \( t \), if \( \hat{a}_t(t) > 0 \), more bikes need to be moved into the docking station \( i \); if \( \hat{a}_t(t) < 0 \), \( i \) can supply other docking stations with bikes; otherwise, \( i \) should not participate in the bike rebalancing operation. Each entry into the bike rebalancing schedule is denoted as a tuple \( (k, i, a) \), which means that \( a \) bikes are to be moved from docking station \( k \) to docking station \( i \) at the current time slot. This bike policy is implemented by Algorithm 1.

**Results and Discussion**

In order to validate STR, we developed a simulator testbed (Figure 1) based on a real-world dataset from London containing a 36-day record of journeys in London bike sharing systems (https://www.kaggle.com/edenau/london-bike-sharing-system-data). It simulates various conditions to study the performance of STR. A video demonstration can be found at https://youtu.be/OGj5z5_EH6A.

The shareable bike usage patterns and the number of bikes involved in STR rebalancing operations over a 24 hour period are shown in Figure 2(a). Two peak demand periods for shareable bikes in London are visible, one at around 9 to 10am, the other around 6 to 8pm. The peaks of rebalancing operations by STR occurred after the end of the morning peak usage period (to satisfy demand during early afternoon), and just before the evening peak period.

**Figure 1: The STR simulation testbed.**

**Figure 2: Experiment Results**

Figure 2(b) shows the demand-supply gap across all docking stations before and after the STR rebalancing operations over a 24 hour period. It can be observed that during day time, STR significantly reduced the gap between demand and supply (in docking stations where demand outstrips supply). Towards the end of the evening peak period, STR stopped rebalancing as it expects that the demand-supply gap will disappear through normal usage towards midnight.

STR offers an effective mechanism for domain experts to transfer their knowledge and preferences to the AI algorithm, without having to rely on learning through trial-and-error. The resulting bike rebalancing plans can be readily explained to users to enhance transparency (Yu et al. 2018).

**Acknowledgements**

This research is supported by Nanyang Assistant Professorship (NAP), AISP-GC-2019-003, NRF-NRFI05-2019-0002, NTU-SDU-CFAIR (NSC-2019-011), and Alibaba-NTU-AIR2019B1.

**References**

