Algorithmic Solutions for Re-Balancing in Bike Sharing: Challenges and Opportunities

Jie Wu

Center for Networked Computing

Temple University

Road Map

1. Introduction
2. Four System Components
3. Re-balancing Through Trucks
4. Re-balancing Through Workers
5. Spatial and Temporal Complexity
6. Challenges and Opportunities
7. Conclusion

1. Introduction

- Smart City
  - Collection of data
  - Management of assets, resources, and services
- Scope
  - Transportation
  - Power plants
  - Utilities
  - Water supply
  - Crime detection
  - School
  - Libraries
  - Hospitals
  - …

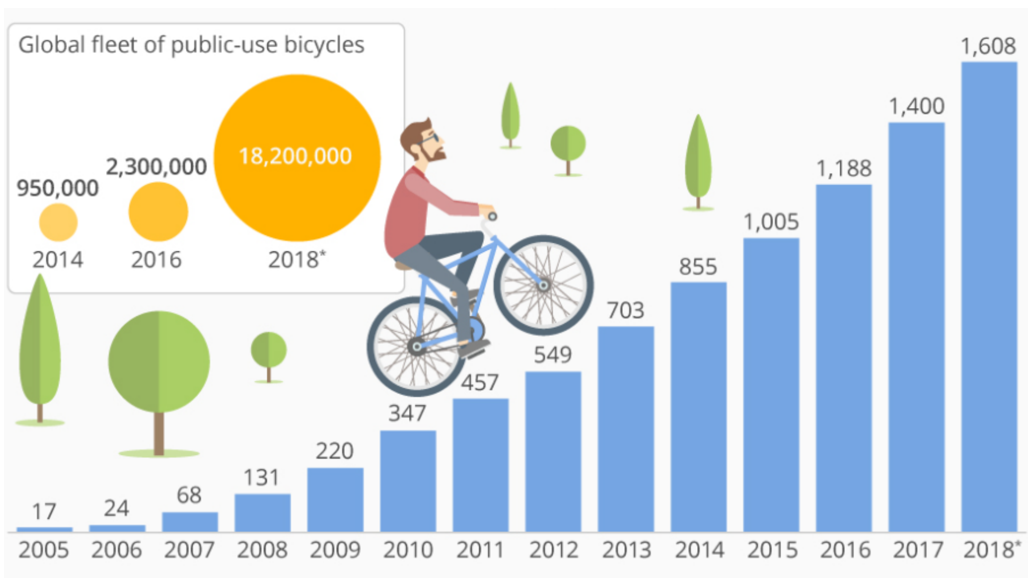
Bike Sharing System (BSS)

BSS

- First/last mile connection
- Rent-Ride-Return
- > 1600 BSSs in > 50 countries

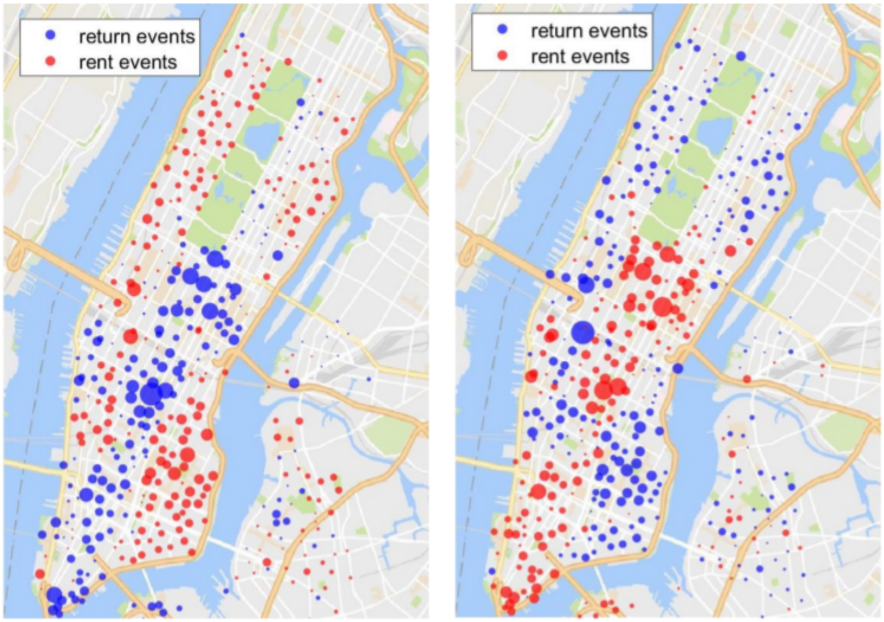
Benefits

- Healthy lifestyle
- Green transportation
- 40% of BSS users drive less



Unbalanced Usage in BSS

- Unbalanced usage
  - Time
  - Space
- Capacity
  - Underflow (full stations)
  - Overflow (empty stations)



(a) AM rush hours: 8:00 - 10:00 AM (b) PM rush hours: 5:00 - 7:00 PM

Re-Balancing in BSS

Dock BSS

- Citi Bike (NYC), Indego (Philly), and GoBike (Bay Area), …
- BikeMi (Milan), Bubi (Budapest)

Dock-less BSS

- ofo and Mobike (in China)
- U-Bicycle and OV-fiets (Europe)
- LimeBike and JUMP (US)

Re-balancing

- Via truck
- Via worker

2. Four System Components

1. System design

- Station number, location, capacity, and bike number
- Facility location problem: area best for placing a station?

2. System prediction

- Mobility modeling
- Traffic prediction

3. System balancing

- Dedicated truck service
- Incentive-based worker recruitment
- Route planning and scheduling

4. Trip advisor

- User guidance
- Re-balance via suggestions

3. Re-balancing Through Trucks

Hamiltonian circle

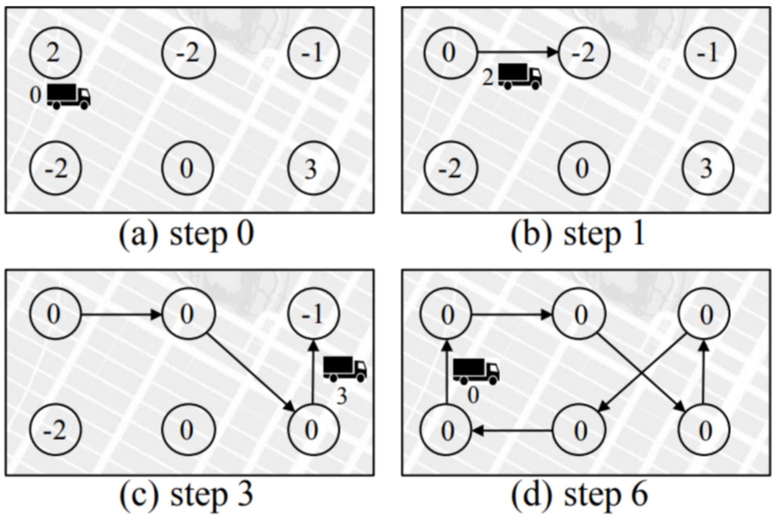
- Trucks move around stations to pick-up/drop-off bikes

Legitimate circle

- Alternating positive pieces and negative pieces s.t. capacity l

Notation

- +m: overflow by m
- -m: underflow by m
- l: truck capacity
- -l ≤ m ≤ l



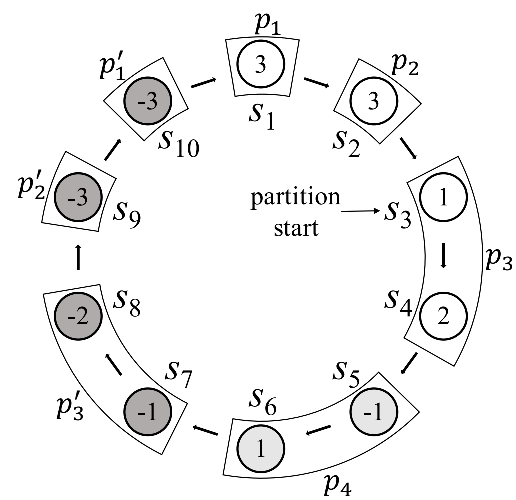
MATCH and GREED Methods

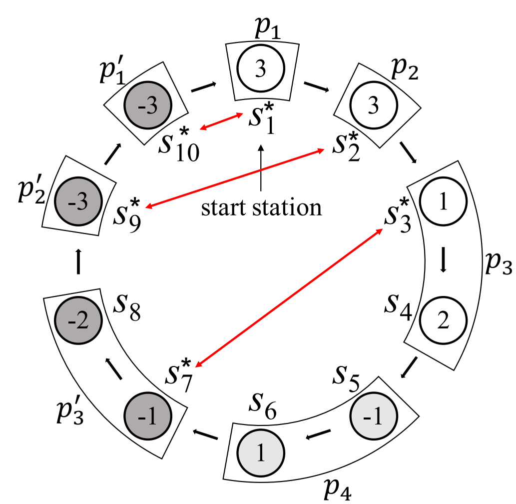
Assumptions

- Predefined Hamiltonian cycle
- Piece length limit: l'

MATCH method

- l': $l/2$, complexity: $O(n^3)$, bound: 6.5
- Min-weight perfect matching: pos., neg., and zero pieces
- Visit each pair following the seq. in the cycle





(1, 3, 7, 8, 4, 5, 6, 9, 2, 10, 1)

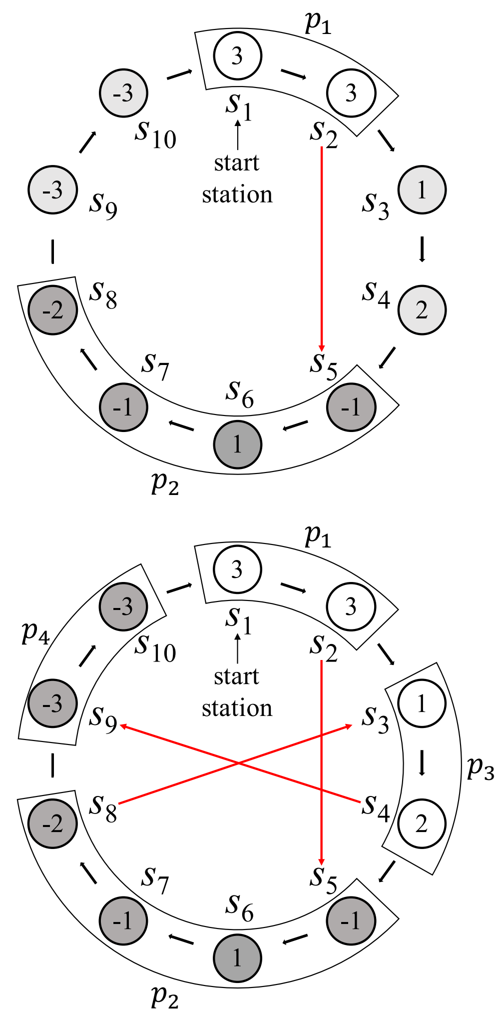
MATCH and GREED Methods

Assumptions

- Predefined Hamiltonian cycle
- Piece length limit: l'

GREED method

- l': l, complexity: $O(n^2)$
- Alternating pos. and neg. following the cycle



(1, 2, 5, 6, 7, 8, 3, 4, 9, 10, 1)

HYBRID Methods

MATCH

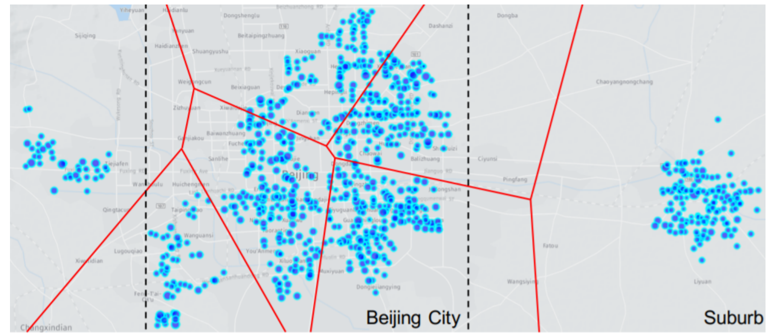
- Sparse mode (primary)
- Small geo-area (secondary)

GREED

- Dense model (primary)
- Large geo area (secondary)

HYBRID

- Two-level hierarchy
- MATCH for intra-cluster
- GREED for inter-cluster



(a) A sample distribution of dock stations in Beijing [26]

	MATCH	GREED	HYBRID
City	2.064	1.108	0.881
City+Suburb	3.016	1.923	1.080
City (Sparse)	1.435	1.781	1.342
City + Suburb (Sparse)	2.597	2.575	1.827

(b) MATCH, GREED, vs HYBRID

M. Charikar et al, Algorithms for capacitated vehicle routing, SIAM, 2001

Y. Duan, J. Wu, and H. Zheng, A greedy approach for vehicle routing, IEEE GLOBECOM, 2018

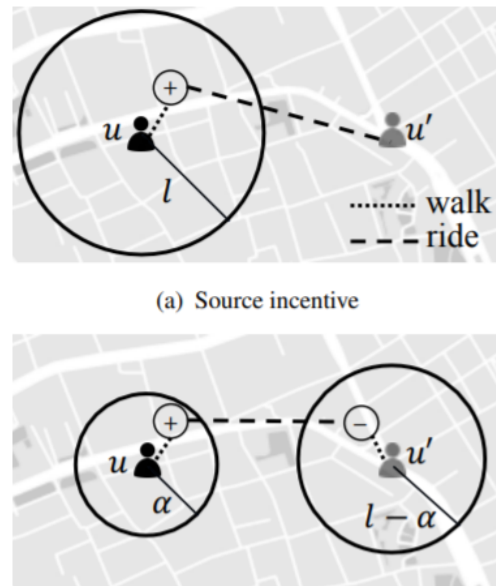
4. Re-balancing Through Workers

Through incentive

- Workers are BSS users
- Overflow: + and underflow: -
- Monetary award prop.to distance

Dock-less incentive

- Source detour bounded by l
- Extensions with detour at both source and destination



(a) Source incentive

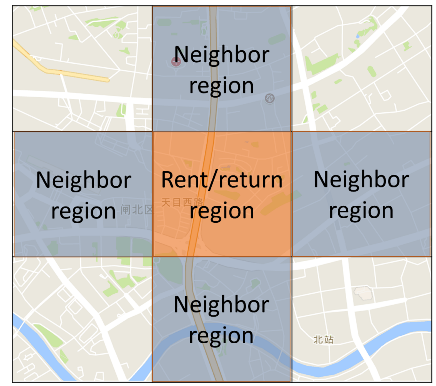
(b) Source and destination incentive

Y. Duan and J. Wu, Optimizaing Rebalance Scheme for Dock-less Bike Sharing Systems with Adaptive User Incentive, IEEE MDM, 2019

Incentive Simulation

Cost of detour δ

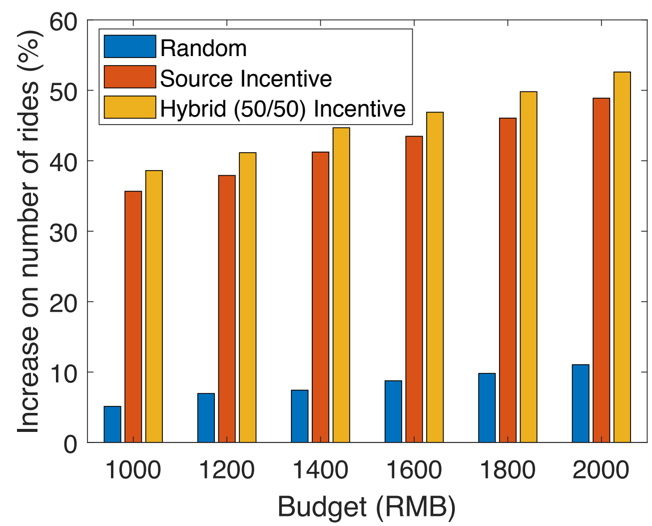
- 0 in original rent/return region
- $\eta\delta^2$ in neighbor regions
- $+\infty$ otherwise



Mobike trace data

Incentive

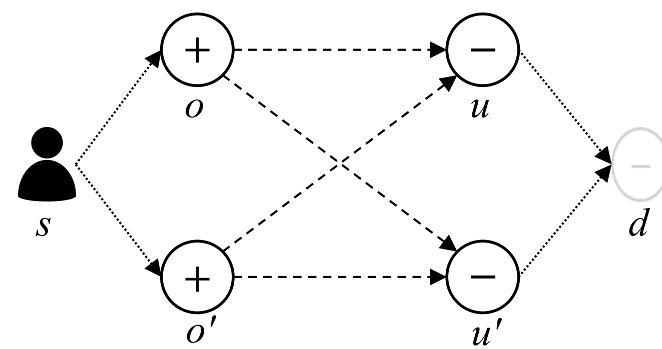
- Learn optimal prizing from usage dynamics
- Higher rent (return) incentive in overflow (underflow) regions



A Global Incentive Approach

Incentive

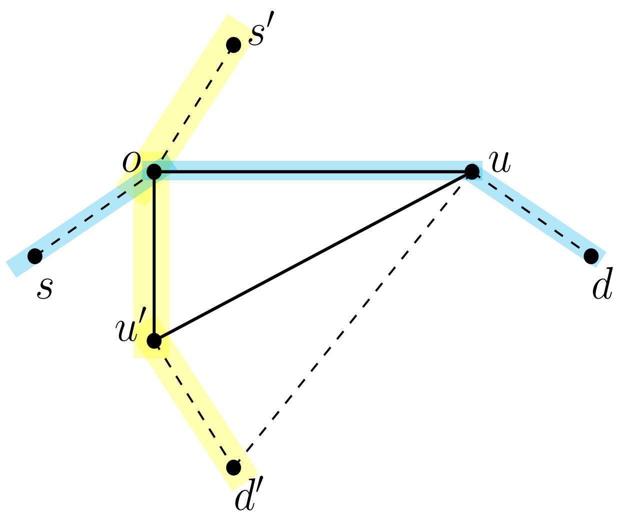
- For both dock and dock-less
- Deal with multiple workers
- 3-D perfect matching
  - Match overflow station with underflow station
  - Match users with a station pair



Y. Duan and J. Wu, Optimizing the crowdsourcing-based bike rebalancing scheme, IEEE ICDCS, 2019

Approximation

- 3-approximation
- Proof sketch:



Optimality of the two rounds of matching

$$\Sigma ou \leq \Sigma ou'$$

$$\Sigma(so + ud) \leq \Sigma(s'o + ud')$$

Triangle inequality

$$\Sigma ud' \leq \Sigma(uu' + u'd')$$

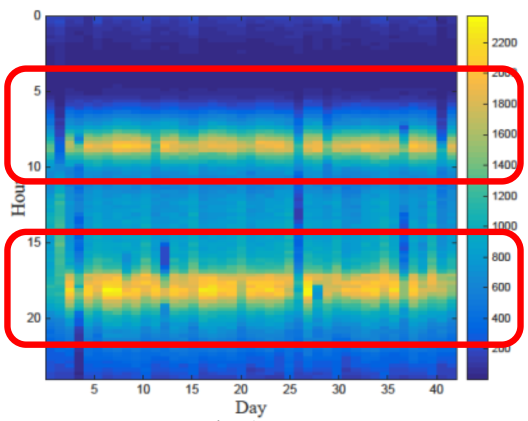
$$\Sigma uu' \leq \Sigma(ou + ou')$$

Combining

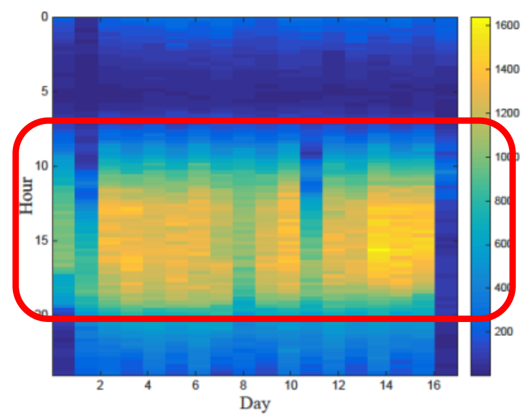
$$\Sigma(so + ou + ud) \leq \Sigma(s'o + 3ou' + u'd') \leq 3OPT$$

5. Spatial and Temporal Complexity

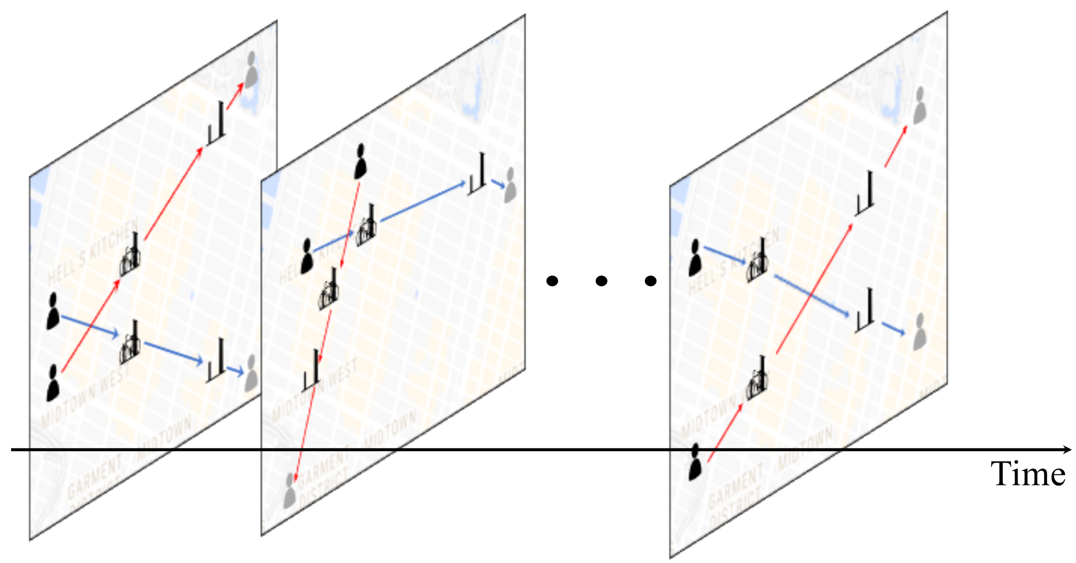
Traffic dynamic: NYC Citi Bike dataset



weekdays



weekends



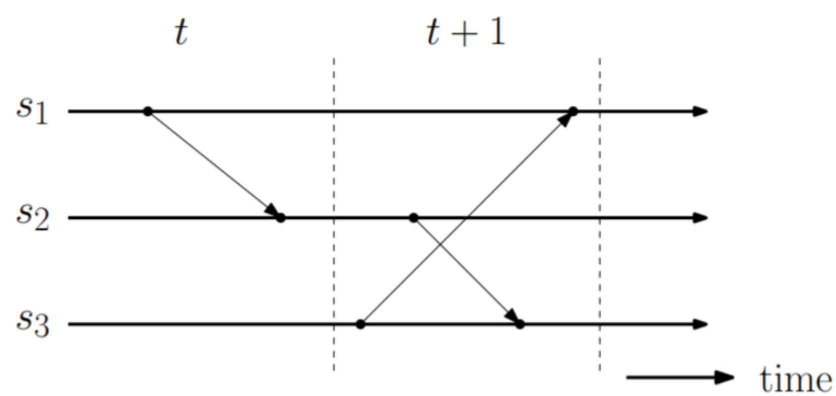
Time-Space View

View

- Horizontal line
  - Status of local station
- Vertical dotted line (slice)
  - Time period between two slices
- Slanted arrow
  - Re-balancing event

Global state

- Local state
- Transition state
- Cut: an event goes across two slices



Frequency Reduction via Look-Ahead

K-hop look ahead

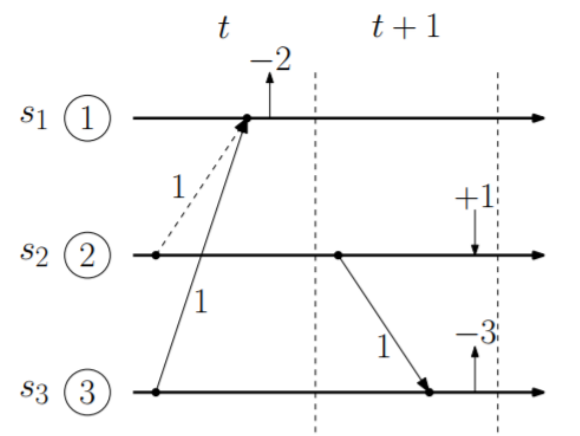
- Once done in the current slice, it can last at least k hops

Greedily look ahead

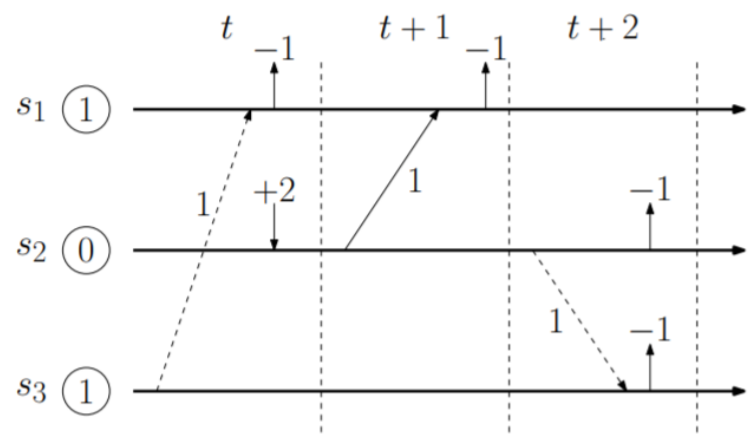
- Uses look-ahead data to make a target move so that the target configuration can last the longest

Greedily look and act ahead

- Cut ahead in any of next k slices (instead of just the current slice)



(a) An example of 2-hop look ahead outperforming 1-hop look ahead



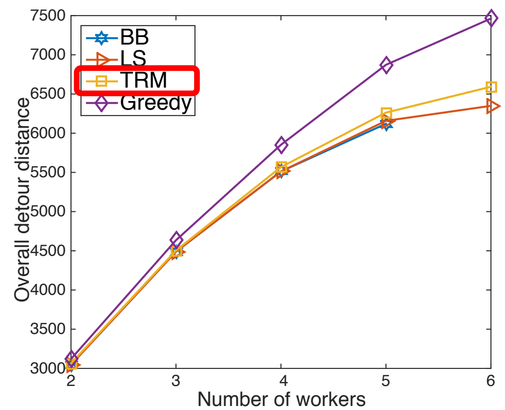
(b) An example of 1-hop look ahead outperforming greedily look ahead

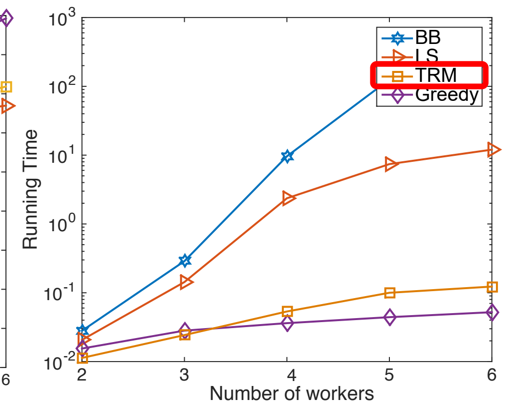
Spatial and Temporal Domain Simulation

Spatial domain

- On a single time slot
- Given rebalance targets
- Minimize worker detour

(BB: Branch & Bound , LS: Local Search, TRM: 2-Round Matching)

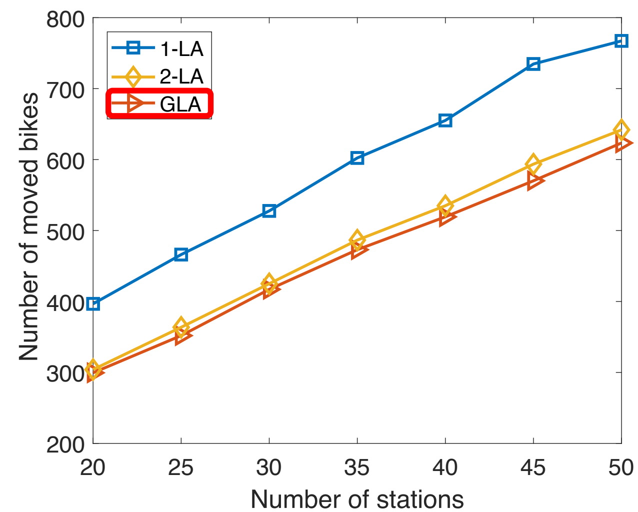




Temporal domain

- Over multiple time slots
- Minimize bike usage

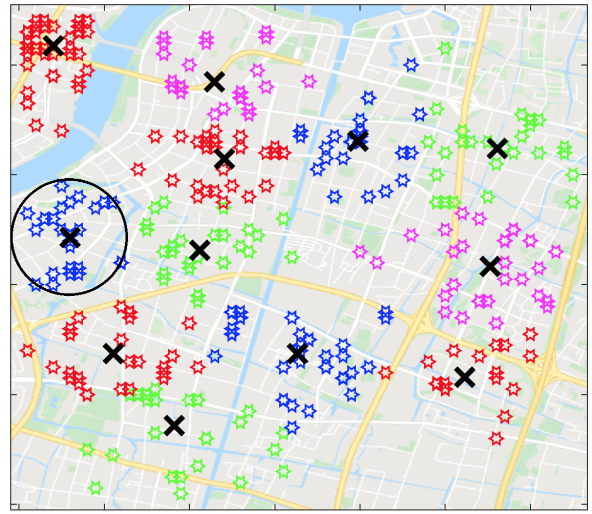
(1-LA: 1-hop, 2-LA: 2-hop, GLA: Greedily)



Extension to Dock-less Scenario

Virtual stations (VS)

- Mesh grid
- K-means
- Density-Based Clustering



Mobike Shanghai Dataset (08/01/16-08/31/16)

Rebalancing VS

- Pick-up
  - nearest in starting VS
- Drop-off
  - nearest in destination VS

6. Challenges and Opportunities

Model extensions

- Models with "cut"
- Capacities for trucks and workers

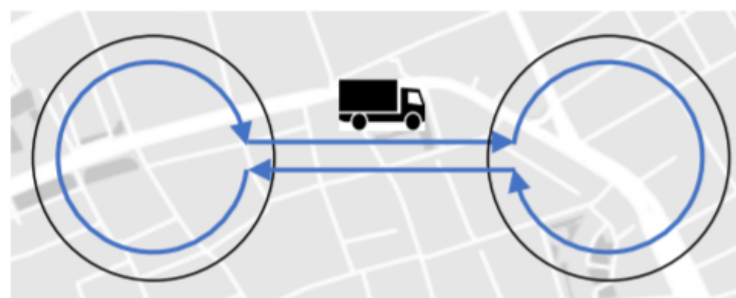
Scalable design

- Geometric partitioning
- Clustering (k-means)

Other models

- Bike recycling
- Usage balance
- Pricing (mechanism design)

(a) Two individual circles

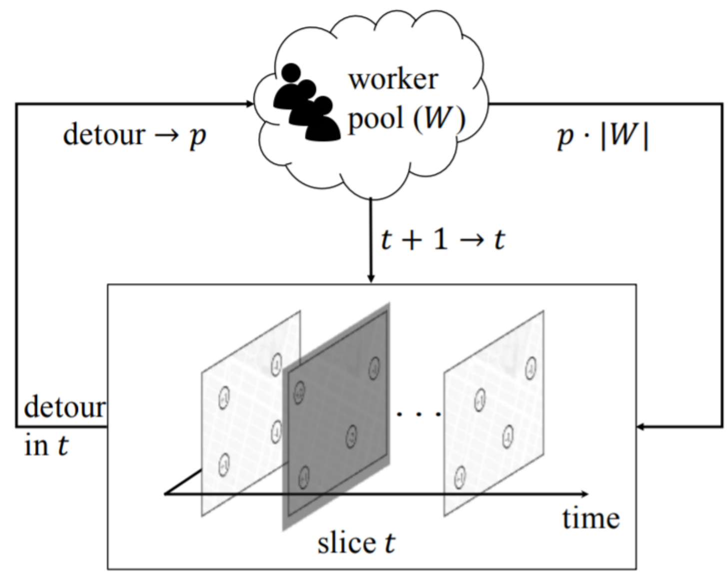


(b) One merged circle

Challenges and Opportunities (Cont'd)

Gaming and Incentive

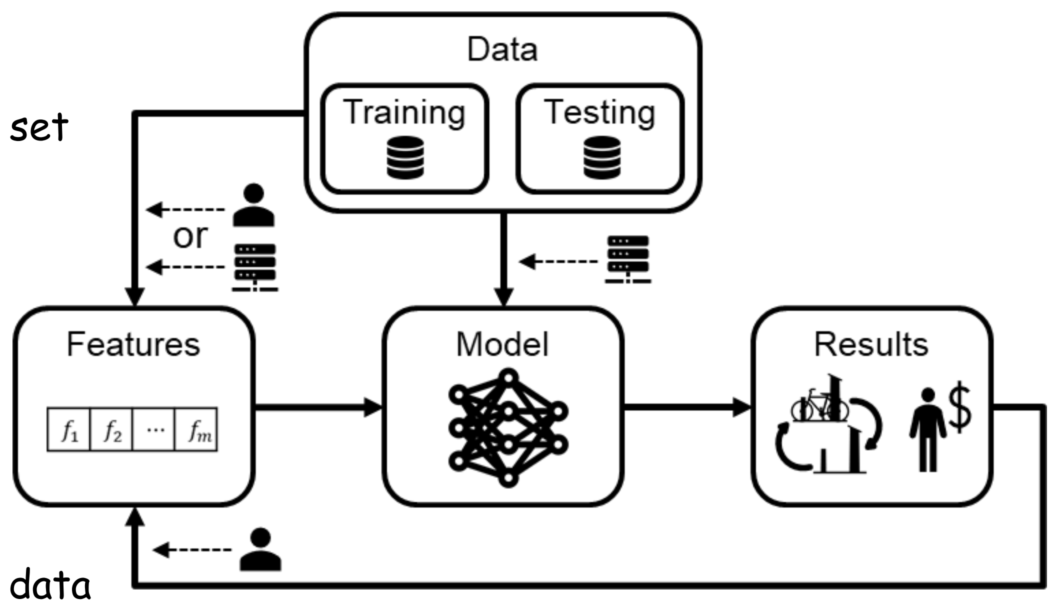
- Stakelberg game
  - BBS operators and workers
- Nash equilibrium
  - Subgames between homo-/hetero workers
- Incentive
  - Reinforcement incentive



Challenges and Opportunities (Cont'd)

ML with data analytics

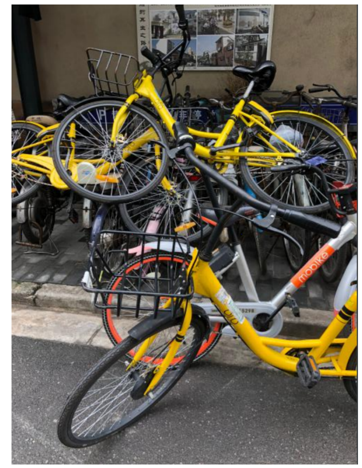
- Effectiveness
  - With the support of a large data set
  - Challenges: biased samples, data sparsity, data missing
- Explainable AI
  - Hybrid approaches
- Robustness
  - Performance deviation due to the data perturbation



Challenges and Opportunities (Cont'd)

Dock vs. dock-less BSS

- Flexibility
- Manageability



Trends

- Dock-less BSSs have appeared largely in US
- Ofo, the largest dock-less BSS in China, suffered financially

A Bigger Picture: Classification

Active transportation

- Fixed (subway, bus, auto-shuttle)
- On-demand (taxi, Uber, DiDi, Lift)
- Hybrid (restricted on-demand)

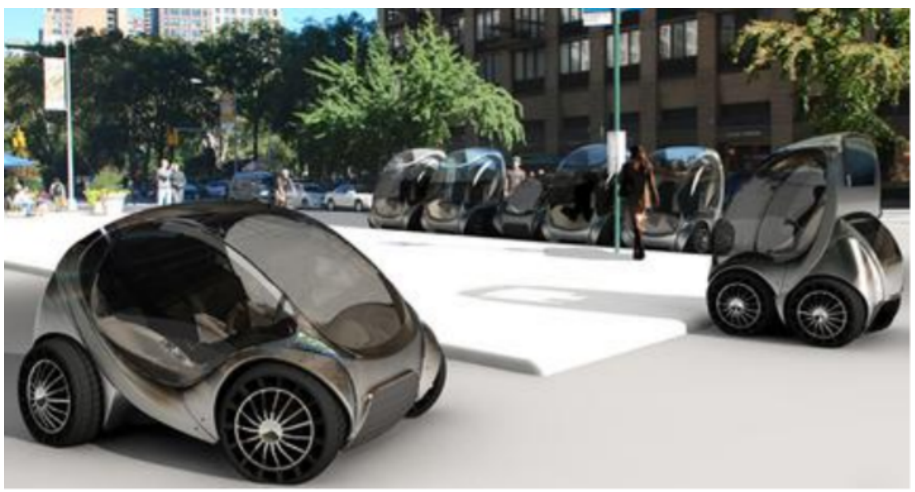
Passive transportation

- ZipCar (first/last ten-mile)
- Bike/e-bike (first/last mile)
- Scooter/e-scooter (first/last mile)

A Bigger Picture: Future of BSSs

Future

- E-bike or two-wheeled e-scooters
- Disappearing dock-less BSS in US



Policy

- Shared responsibility
  - Credit systems
- Safety and regulation
  - Sidewalk, bike lanes, and car lanes
  - Scooter: sidewalk or bike lane?
  - How about the folded-mini cars (in MIT's CityCar Project)?

7. Conclusions

- Bike Sharing Systems (BSSs)
  - Bike re-balancing issue
- Solutions
  - Algorithmic solutions
  - ML solutions with data analytics
- Future of BSSs
  - Policies and regulations
  - Role in a smart-city ecosystem