

Lecture Notes in Mobility

Jan Brinkmann

# Active Balancing of Bike Sharing Systems

 Springer

# **Lecture Notes in Mobility**

## **Series Editor**

Gereon Meyer, VDI/VDE Innovation und Technik GmbH, Berlin, Germany

More information about this series at <http://www.springer.com/series/11573>

Jan Brinkmann

# Active Balancing of Bike Sharing Systems

 Springer

Jan Brinkmann  
Institut für Wirtschaftsinformatik  
Technische Universität Braunschweig  
Braunschweig, Germany

ISSN 2196-5544                      ISSN 2196-5552 (electronic)  
Lecture Notes in Mobility  
ISBN 978-3-030-35011-6              ISBN 978-3-030-35012-3 (eBook)  
<https://doi.org/10.1007/978-3-030-35012-3>

© Springer Nature Switzerland AG 2020

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Switzerland AG  
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

# Foreword

Vehicle sharing has received a remarkable attention as a new means of urban transportation. Practice has shown that the one-way use of vehicles follows mobility patterns of people leading to temporal and spatial imbalances with respect to the distribution of vehicles in the city. In station-based bike sharing systems, customers suffer from the absence of bikes in case of a potential rental and the absence of bike racks in the case of a bike return. Station-less systems have claimed to offer flexibility; however, they have failed to overcome the deficiency of bike imbalances.

System operators see the requirement of redistributing bikes between city areas over the day at significant expenses. A methodological support of bike logistics has concentrated on static optimization models. These models are typically fed with data of historic bike usage. Since history does not repeat itself, optimal solutions obtained from static model cannot be implemented due to stochastics with respect to actual bike usage.

Jan Brinkmann focuses on a control approach deciding dynamically about bike imbalances to be resolved. He combines control with an anticipation of future redistribution demand by means of online simulation. The simulation takes into account the driving time needed to arrive at the respective station, the loading or unloading time at this station as well as the avoidance of future fails resulting from bike inventory changes.

The informative value of the simulation strongly depends on the simulation horizon. A short horizon may not reflect the utility of the station visit. Simulating over a long horizon may report on customer fails, which no longer relate to the respective station visit. Jan Brinkmann is able to provide evidence that a suitable simulation horizon is by no means fixed, but depends on the particular situation, i.e., the time of day. To this end, he develops an approximate dynamic programming approach determining heterogeneous simulation horizons iteratively.

The above consideration applies to the one vehicle case only. Whenever a fleet of trucks is employed for bike redistribution, the decentral decisions of the trucks are no longer independent of each other. Since all of them follow the same decision model, it may happen that demanding stations may accidentally be visited multiple times. Jan Brinkmann suggests different levels of coordination coming along with a

slightly growing need for information exchange. The trucks operate independently of each other and take decision for their own operation. Like in the one vehicle case before, decisions comprise the number of bikes to be loaded or unloaded at the current station and the station to be visited next.

The control approaches developed are carefully validated for real-world instances of bike sharing systems. Promising results are obtained for all instances considered. In particular, the approach is best suited for bike sharing systems which do not show a regular structure of bike imbalances due to commuter travel. Regular flows from residential areas to office districts in the morning and reverse flows in the late afternoon are relatively easy to predict and to counteract. More challenging are complex mobility patterns consisting of mixed work, shopping, and leisure activities. Results obtained indicate that these complex interactions can be supported much better by control than by static optimization.

Jan Brinkmann pioneers online control models for the redistribution logistics of bike sharing systems. The work bases on a solid understanding of bike sharing system, business models, and related activities. The control approach pursued has been well received by the transportation research community as well as by colleagues working in Operations Research. This book summarizes research of recent years by giving a comprehensive introduction into control approaches for today's and forthcoming vehicle sharing systems.

Braunschweig, Germany  
January 2019

Dirk C. Mattfeld

# Preface

Many cities suffer from discomforts caused by individual and motorized traffic. Therefore, city administrations implement sustainable shared mobility services such as bike sharing systems (BSSs). In BSSs, users are allowed to rent and return bikes on short notice at stations. Data analysis reveals that rental and return requests follow spatio-temporal patterns such as commuter usage and leisure activities. In the morning, commuter usage is indicated by mainly rental requests in residential areas and mainly return requests in working areas. This behavior inverts in the course of the day. The resulting unequal requests lead stations to become empty or full. Requests to rent bikes will fail at empty stations. At full stations, requests to return bikes will fail.

Providers counteract these inconveniences by means of balancing. In this work, we focus on the operational management's view on the balancing of BSSs. That is, the provider schedules transport vehicles relocating bikes between stations in order to minimize the amount of failed requests. As requests are uncertain, the resulting challenge is to identify stations with a lack or a surplus of bikes. To this end, we introduce approaches simulating future requests and approximating expected amounts of failed requests. Then, anticipation is enabled by means of including the approximations in the decision making process.

We evaluate our approaches in case studies based on real-world data. The results point out that our approaches are able to reduce the amount of failed requests significantly compared to common benchmarks from literature.

Braunschweig, Germany  
January 2019

Jan Brinkmann



# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>Part I Preliminaries</b>		
<b>2</b>	<b>Bike Sharing Systems</b>	<b>7</b>
2.1	Urban Mobility	7
2.2	Benefits	8
2.2.1	Reduction of Traffic	8
2.2.2	Improvement of Health	9
2.2.3	Increase in Tourists Attractiveness	9
2.3	Functionality	10
2.3.1	Free-Floating	10
2.3.2	Station-Based	10
2.4	Request Patterns	11
2.4.1	Seasons and Weather	12
2.4.2	Commuters	12
2.4.3	Leisure and Tourists	12
2.5	Management Layers	13
2.5.1	Strategical Management	13
2.5.2	Tactical Management	15
2.5.3	Operational Management	17
<b>3</b>	<b>Optimization Problems</b>	<b>19</b>
3.1	Vehicle Routing	19
3.1.1	Traveling Salesman Problem	19
3.1.2	Capacitated Vehicle Routing Problem	20
3.1.3	Vehicle Routing Problem with Time Windows	20
3.1.4	Pickup-and-Delivery Problem	20
3.1.5	Inventory Routing Problem	20

3.2	Inventory Routing for Bike Sharing Systems	21
3.2.1	No Request	22
3.2.2	Request	23
<b>4</b>	<b>Dynamic Decision Making</b>	<b>31</b>
4.1	Markov Decision Processes	31
4.2	Approximate Dynamic Programming	35
4.2.1	Myopic	36
4.2.2	Lookahead	36
4.2.3	Value Function Approximation	38
<b>Part II Application</b>		
<b>5</b>	<b>The Stochastic-Dynamic Multi-Vehicle Inventory Routing Problem for Bike Sharing Systems</b>	<b>43</b>
5.1	Narrative	43
5.2	Infrastructure	44
5.3	Markov Decision Process	44
5.4	Example	47
5.5	Challenges	48
<b>6</b>	<b>Lookahead Policies</b>	<b>51</b>
6.1	Outline	51
6.2	Definition	53
6.2.1	Simulation	53
6.2.2	Optimization	57
6.3	Algorithms	64
6.3.1	Lookahead Policy	64
6.3.2	Online Simulations	66
6.3.3	Offline Simulations	66
6.3.4	Matrix Maximum Algorithm	67
<b>7</b>	<b>Dynamic Lookahead Horizons</b>	<b>69</b>
7.1	Outline	69
7.2	Definition	71
7.2.1	Sequences of Lookahead Horizons	72
7.2.2	Value Function Approximation	73
7.2.3	Boltzmann Exploration	74
7.3	Algorithms	77
7.3.1	Value Function Approximation	77
7.3.2	Boltzmann Exploration	79

<b>8</b>	<b>Case Studies</b>	81
8.1	Real-World Data	81
8.1.1	Data Preprocessing	81
8.1.2	Resulting Data Set	82
8.2	Instances	83
8.3	Transition	84
8.4	Benchmarks	85
8.4.1	Safety Buffer-Tending Relocation Policy	87
8.4.2	Rollout Algorithms	88
8.5	Parametrization	88
8.5.1	Safety Buffer-Tending Relocations	89
8.5.2	Online Simulations	89
8.5.3	Static Lookahead Policies	91
8.5.4	Dynamic Lookahead Policies	91
8.5.5	Rollout Algorithms	92
8.6	Results	94
8.6.1	The Value of Coordination	95
8.6.2	The Value of Anticipation	95
8.6.3	Individual Results	96
8.7	Analysis	98
8.7.1	Optimal Assignment	98
8.7.2	Learning Curves	100
8.7.3	Dynamic Lookahead Horizons	101
<b>Part III Conclusion</b>		
<b>9</b>	<b>Managerial Implications</b>	109
<b>10</b>	<b>Future Research</b>	111
10.1	Model	111
10.2	Method	114
<b>Appendix A: Parameters</b>		117
<b>Appendix B: Results</b>		125
<b>References</b>		177

# Acronyms

ADP	Approximate Dynamic Programming
AV	Autonomous Vehicle
BSS	Bike Sharing System
DLA	Dynamic Lookahead Policy
DPS	Dynamic Policy Selection
IRP	Inventory Routing Problem
LA	Lookahead Policy
LUT	Lookup Table
MDP	Markov Decision Process
PTS	Public Transport System
SMS	Shared Mobility System
SLA	Static Lookahead Policy
STR	Safety Buffer-tending Relocation Policy
VFA	Value Function Approximation
VRP	Vehicle Routing Problem

# List of Figures

Fig. 1.1	Overview on the parts and chapter of this work . . . . .	4
Fig. 2.1	A station of a bike sharing system . . . . .	11
Fig. 2.2	Overview on the management layers . . . . .	14
Fig. 4.1	A dynamic decision process (adapted, Meisel 2011). . . . .	32
Fig. 4.2	A Markov decision process (adapted, Ulmer et al. 2015). . . . .	32
Fig. 4.3	An exemplary decision tree (adapted, Ulmer 2017) . . . . .	34
Fig. 4.4	A rollout algorithm’s simulations (adapted, Ulmer et al. 2016) . . . . .	37
Fig. 4.5	An exemplary trajectory in a decision tree . . . . .	38
Fig. 4.6	An iteration of an VFA’s approximation phase. . . . .	39
Fig. 5.1	An exemplary MDP of the $IPR_{BSS}$ (adapted, Brinkmann et al. 2019b). . . . .	47
Fig. 6.1	Overview of a lookahead policy (adapted, Brinkmann et al. 2019a). . . . .	52
Fig. 6.2	Three inventory decisions and the resulting fill levels. . . . .	54
Fig. 6.3	An online lookahead’s simulations . . . . .	55
Fig. 6.4	Observed fill levels and failed requests in an exemplary simulation. . . . .	58
Fig. 6.5	Observed rental requests in an exemplary simulation . . . . .	59
Fig. 6.6	Failed requests in an exemplary simulation (Brinkmann et al. 2019a) . . . . .	60
Fig. 6.7	Observed fill level and failed requests in an exemplary simulation (adapted, Brinkmann et al. 2019b). . . . .	61
Fig. 6.8	An exemplary assignment problem (adapted, Brinkmann et al. 2019b). . . . .	62
Fig. 7.1	An iteration of a value function approximation’s approximation phase in combination with a dynamic lookahead policy (adapted, Brinkmann et al. 2019a) . . . . .	70
Fig. 7.2	Probabilities of two exemplary values in the course of an approximation phase . . . . .	76

Fig. 7.3 Probabilities of three exemplary values in the course of an approximation phase . . . . . 77

Fig. 8.1 Temporal distributions of trips (adapted, Brinkmann et al. 2019b) . . . . . 83

Fig. 8.2 Processes between two decision points . . . . . 84

Fig. 8.3 Procedure of handling events (adapted, Brinkmann et al. 2019a) . . . . . 86

Fig. 8.4 Determining the safety buffers of STR . . . . . 89

Fig. 8.5 Determining the number of simulation runs of  $SLA_{on}$ : failed requests. . . . . 90

Fig. 8.6 Determining the number of simulation runs of  $SLA_{on}$ : runtime . . . . . 91

Fig. 8.7 Determining the lookahead horizon of  $SLA_{on}$ . . . . . 92

Fig. 8.8 Determining the lookahead horizon of  $SLA_{off}$  . . . . . 92

Fig. 8.9 Simulation runs of rollout algorithms in Minneapolis: failed requests. . . . . 93

Fig. 8.10 Simulation runs of rollout algorithms in San Francisco: failed requests. . . . . 93

Fig. 8.11 Simulation runs of rollout algorithms in Minneapolis: runtime . . . . . 94

Fig. 8.12 Simulation runs of rollout algorithms in San Francisco: runtime . . . . . 94

Fig. 8.13 The values of coordination . . . . . 95

Fig. 8.14 The values of anticipation. . . . . 96

Fig. 8.15 The results of Minneapolis . . . . . 97

Fig. 8.16 The results of San Francisco. . . . . 97

Fig. 8.17 Improvement ratios of heuristic over optimal assignment. . . . . 99

Fig. 8.18 Failed requests in the first 500 trajectories of an approximation phase. . . . . 100

Fig. 8.19 Failed requests in the course of an approximation phase . . . . . 100

Fig. 8.20 VFA-determined lookahead horizons for one vehicle in Minneapolis and San Francisco . . . . . 101

Fig. 8.21 VFA-determined lookahead horizons for four vehicle in Minneapolis and San Francisco . . . . . 102

Fig. 8.22 A priori DLAs' lookahead horizons . . . . . 103

Fig. 8.23 Ex Post DLAs' lookahead horizons one vehicle in Minneapolis and San Francisco . . . . . 103

Fig. 8.24 Ex Post DLAs' lookahead horizons for four vehicles in Minneapolis and San Francisco . . . . . 104

Fig. 8.25 The results of DLAs with manual-determined horizons in Minneapolis . . . . . 104

Fig. 8.26 The results of DLAs with manual-determined horizons in San Francisco . . . . . 105

# List of Tables

Table 3.1	Literature classification on inventory routing for bike sharing systems: no requests . . . . .	24
Table 3.2	Literature classification on inventory routing for bike sharing systems: requests . . . . .	26
Table 4.1	Overview on categories of approximate dynamic programming . . . . .	35
Table 5.1	Notation of the bike sharing system’s infrastructure . . . . .	44
Table 5.2	Notation of the Markov decision process . . . . .	45
Table 7.1	Notation of the value function approximation . . . . .	71
Table 8.1	Characteristics of investigated bike sharing systems . . . . .	82
Table A.1	Parametrization for STR, $SLA_{on}$ , and $SLA_{off}$ . . . . .	117
Table A.2	Simulation runs for Minneapolis . . . . .	118
Table A.3	Simulation runs for San Francisco . . . . .	119
Table A.4	Dynamic lookahead horizons for $DLA_{on}$ and $DLA_{off}$ . . . . .	120
Table A.5	Dynamic lookahead horizons for manual DLAs . . . . .	121
Table A.6	Results of rollout algorithms in Minneapolis . . . . .	122
Table A.7	Results of rollout algorithms in San Francisco . . . . .	123
Table B.1	Results of STR and one vehicle in Minneapolis . . . . .	125
Table B.2	Results of STR and one vehicle in Minneapolis . . . . .	126
Table B.3	Results of STR and one vehicle in Minneapolis . . . . .	127
Table B.4	Results of $SLA_{on}$ and one vehicle in Minneapolis . . . . .	128
Table B.5	Results of $SLA_{off}$ and one vehicle in Minneapolis . . . . .	129
Table B.6	Results of the DLAs and one vehicle in Minneapolis . . . . .	130
Table B.7	Results of STR, two vehicles, and independent dispatching in Minneapolis . . . . .	130
Table B.8	Results of STR, two vehicles, and heuristic dispatching in Minneapolis . . . . .	130
Table B.9	Results of $SLA_{on}$ , two vehicles, and independent dispatching in Minneapolis . . . . .	131

Table B.10 Results of  $SLA_{on}$ , two vehicles, and heuristic dispatching in Minneapolis. . . . . 132

Table B.11 Results of  $SLA_{on}$ , two vehicles, and optimal dispatching in Minneapolis. . . . . 133

Table B.12 Results of  $SLA_{off}$ , two vehicles, and independent dispatching in Minneapolis . . . . . 134

Table B.13 Results of  $SLA_{off}$ , two vehicles, and heuristic dispatching in Minneapolis. . . . . 135

Table B.14 Results of  $SLA_{off}$ , two vehicles, and optimal dispatching in Minneapolis. . . . . 136

Table B.15 Results of the DLAs and two vehicles in Minneapolis . . . . . 137

Table B.16 Results of STR, three vehicles, and independent dispatching in Minneapolis . . . . . 137

Table B.17 Results of STR, three vehicles, and heuristic dispatching in Minneapolis. . . . . 137

Table B.18 Results of  $SLA_{on}$ , three vehicles, and independent dispatching in Minneapolis . . . . . 138

Table B.19 Results of  $SLA_{on}$ , three vehicles, and heuristic dispatching in Minneapolis. . . . . 139

Table B.20 Results of  $SLA_{on}$ , three vehicles, and optimal dispatching in Minneapolis. . . . . 140

Table B.21 Results of  $SLA_{off}$ , three vehicles, and independent dispatching in Minneapolis . . . . . 141

Table B.22 Results of  $SLA_{off}$ , three vehicles, and heuristic dispatching in Minneapolis. . . . . 142

Table B.23 Results of  $SLA_{off}$ , three vehicles, and optimal dispatching in Minneapolis. . . . . 143

Table B.24 Results of the DLAs and three vehicles in Minneapolis . . . . . 144

Table B.25 Results of STR, four vehicles, and independent dispatching in Minneapolis. . . . . 144

Table B.26 Results of STR, four vehicles, and heuristic dispatching in Minneapolis. . . . . 144

Table B.27 Results of  $SLA_{on}$ , four vehicles, and independent dispatching in Minneapolis . . . . . 145

Table B.28 Results of  $SLA_{on}$ , four vehicles, and heuristic dispatching in Minneapolis. . . . . 146

Table B.29 Results of  $SLA_{on}$ , four vehicles, and optimal dispatching in Minneapolis. . . . . 147

Table B.30 Results of  $SLA_{off}$ , four vehicles, and independent dispatching in Minneapolis . . . . . 148

Table B.31 Results of  $SLA_{off}$ , four vehicles, and heuristic dispatching in Minneapolis. . . . . 149

Table B.32 Results of  $SLA_{off}$ , four vehicles, and optimal dispatching in Minneapolis. . . . . 150



Table B.33 Results of the DLAs and four vehicles in Minneapolis . . . . . 151

Table B.34 Results of STR and one vehicle in San Francisco . . . . . 152

Table B.35 Results of  $SLA_{on}$  and one vehicle in San Francisco . . . . . 153

Table B.36 Results of  $SLA_{off}$  and one vehicle in San Francisco . . . . . 154

Table B.37 Results of the DLAs and one vehicle in San Francisco . . . . . 155

Table B.38 Results of STR, two vehicles, and independent  
dispatching in San Francisco . . . . . 155

Table B.39 Results of STR, two vehicles, and independent dispatching  
in San Francisco . . . . . 155

Table B.40 Results of  $SLA_{on}$ , two vehicles, and independent dispatching  
in San Francisco . . . . . 156

Table B.41 Results of  $SLA_{on}$ , two vehicles, and heuristic dispatching  
in San Francisco . . . . . 157

Table B.42 Results of  $SLA_{on}$ , two vehicles, and optimal dispatching  
in San Francisco . . . . . 158

Table B.43 Results of  $SLA_{off}$ , two vehicles, and independent dispatching  
in San Francisco . . . . . 159

Table B.44 Results of  $SLA_{off}$ , two vehicles, and heuristic dispatching  
in San Francisco . . . . . 160

Table B.45 Results of  $SLA_{off}$ , two vehicles, and optimal dispatching  
in San Francisco . . . . . 161

Table B.46 Results of the DLAs and two vehicles in San Francisco . . . . . 162

Table B.47 Results of STR, three vehicles, and independent dispatching  
in San Francisco . . . . . 162

Table B.48 Results of STR, three vehicles, and heuristic dispatching  
in San Francisco . . . . . 162

Table B.49 Results of  $SLA_{on}$ , three vehicles, and independent  
dispatching in San Francisco . . . . . 163

Table B.50 Results of  $SLA_{on}$ , three vehicles, and heuristic dispatching  
in San Francisco . . . . . 164

Table B.51 Results of  $SLA_{on}$ , three vehicles, and optimal dispatching  
in San Francisco . . . . . 165

Table B.52 Results of  $SLA_{off}$ , three vehicles and independent  
dispatching in San Francisco . . . . . 166

Table B.53 Results of  $SLA_{off}$ , three vehicles, and heuristic dispatching  
in San Francisco . . . . . 167

Table B.54 Results of  $SLA_{off}$ , three vehicles, and optimal dispatching  
in San Francisco . . . . . 168

Table B.55 Results of the DLAs and three vehicles in San Francisco . . . . . 169

Table B.56 Results of STR, four vehicles, and independent dispatching  
in San Francisco . . . . . 169

Table B.57 Results of STR, four vehicles, and heuristic dispatching  
in San Francisco . . . . . 169

Table B.58	Results of $SLA_{on}$ four vehicles, and independent dispatching in San Francisco . . . . .	170
Table B.59	Results of $SLA_{on}$ , four vehicles, and heuristic dispatching in San Francisco . . . . .	171
Table B.60	Results of $SLA_{on}$ , four vehicles, and optimal dispatching in San Francisco . . . . .	172
Table B.61	Results of $SLA_{off}$ , four vehicles, and independent dispatching in San Francisco . . . . .	173
Table B.62	Results of $SLA_{off}$ , four vehicles, and heuristic dispatching in San Francisco . . . . .	174
Table B.63	Results of $SLA_{off}$ , four vehicles, and optimal dispatching in San Francisco . . . . .	175
Table B.64	Results of the DLAs and four vehicles in San Francisco. . . . .	176

# List of Algorithms

Algorithm 1	Lookahead Policy . . . . .	65
Algorithm 2	Online Simulations . . . . .	65
Algorithm 3	Offline Simulations . . . . .	66
Algorithm 4	Matrix Maximum Algorithm . . . . .	67
Algorithm 5	Value Function Approximation . . . . .	78
Algorithm 6	Boltzmann Exploration . . . . .	79

# Chapter 1

## Introduction



Increasing urbanization and mobility demand lead to a large volume of traffic in urban areas. As the traffic infrastructure is limited, too much traffic results in traffic jams, noise, and air pollution. City administrations focus on a reduction of the individual traffic to tackle these discomforts. Therefore, collective traffic modes, i.e., public transport systems (PTSs), are launched or expanded. Conventional modes are buses and trams. On the one hand side, PTSs may be able to reduce urban traffic. But on the other hand side, the comfort is reduced. First, the users' actual origins and destinations are not necessarily in walking distance to a bus or tram station. Second, buses and trams are often delayed or crowded.

Shared mobility services (SMSs), such as car, bike, and scooter sharing systems, are promising alternatives as well as complements to conventional PTS. An SMS grants the access to an available wheeler where the user does not become the owner. The access is granted for one trip, i.e., for the time span the user needs to drive from his origin to his destination. A trip can be started any time a car, bike, or scooter is available without restrictions due to timetables. When the user ends his trip, it becomes available to other users. In this way, car sharing systems can reduce the number of cars in the city significantly (Archer 2017). Further, bike sharing systems (BSSs) offer emission-free and sustainable transport. In free-floating BSSs, bikes are distributed in the operation area. In station-based BSSs, the access to bikes is granted at stations, i.e., bikes are rented and returned at predefined locations. Stations are capacitated, i.e., a limited number of bike racks is available. The network of stations expands over the city center as well as residential areas. Thus, users can satisfy their mobility demand completely with the BSS if origin and destination are in cycle distance. BSSs serve as a complement to PTS if the next PTS stations is not in walking distance. Then, a user can rent a bike near his origin and can cycle to the next PTS station (Lin and Yang 2011).

Information technology found its way into BSSs to support user authentications and payments, and to record rental and return requests (Vogel 2016). The resulting availability of data allows data analysis to reveal spatio-temporal patterns of requests. BSSs are often used by commuters. Accordingly, in the morning, commuters request to rent bikes in residential areas and request to return bikes in working areas. In