



University Transportation Research Center - Region 2

# Final Report

## Real-time Dynamic Pricing for Bicycle Sharing Programs

---

Performing Organization: University at Buffalo (SUNY)



October 2014

Sponsor:  
University Transportation Research Center - Region 2

## University Transportation Research Center - Region 2

The Region 2 University Transportation Research Center (UTRC) is one of ten original University Transportation Centers established in 1987 by the U.S. Congress. These Centers were established with the recognition that transportation plays a key role in the nation's economy and the quality of life of its citizens. University faculty members provide a critical link in resolving our national and regional transportation problems while training the professionals who address our transportation systems and their customers on a daily basis.

The UTRC was established in order to support research, education and the transfer of technology in the field of transportation. The theme of the Center is "Planning and Managing Regional Transportation Systems in a Changing World." Presently, under the direction of Dr. Camille Kamga, the UTRC represents USDOT Region II, including New York, New Jersey, Puerto Rico and the U.S. Virgin Islands. Functioning as a consortium of twelve major Universities throughout the region, UTRC is located at the CUNY Institute for Transportation Systems at The City College of New York, the lead institution of the consortium. The Center, through its consortium, an Agency-Industry Council and its Director and Staff, supports research, education, and technology transfer under its theme. UTRC's three main goals are:

### Research

The research program objectives are (1) to develop a theme based transportation research program that is responsive to the needs of regional transportation organizations and stakeholders, and (2) to conduct that program in cooperation with the partners. The program includes both studies that are identified with research partners of projects targeted to the theme, and targeted, short-term projects. The program develops competitive proposals, which are evaluated to insure the most responsive UTRC team conducts the work. The research program is responsive to the UTRC theme: "Planning and Managing Regional Transportation Systems in a Changing World." The complex transportation system of transit and infrastructure, and the rapidly changing environment impacts the nation's largest city and metropolitan area. The New York/New Jersey Metropolitan has over 19 million people, 600,000 businesses and 9 million workers. The Region's intermodal and multimodal systems must serve all customers and stakeholders within the region and globally. Under the current grant, the new research projects and the ongoing research projects concentrate the program efforts on the categories of Transportation Systems Performance and Information Infrastructure to provide needed services to the New Jersey Department of Transportation, New York City Department of Transportation, New York Metropolitan Transportation Council, New York State Department of Transportation, and the New York State Energy and Research Development Authority and others, all while enhancing the center's theme.

### Education and Workforce Development

The modern professional must combine the technical skills of engineering and planning with knowledge of economics, environmental science, management, finance, and law as well as negotiation skills, psychology and sociology. And, she/he must be computer literate, wired to the web, and knowledgeable about advances in information technology. UTRC's education and training efforts provide a multidisciplinary program of course work and experiential learning to train students and provide advanced training or retraining of practitioners to plan and manage regional transportation systems. UTRC must meet the need to educate the undergraduate and graduate student with a foundation of transportation fundamentals that allows for solving complex problems in a world much more dynamic than even a decade ago. Simultaneously, the demand for continuing education is growing – either because of professional license requirements or because the workplace demands it – and provides the opportunity to combine State of Practice education with tailored ways of delivering content.

### Technology Transfer

UTRC's Technology Transfer Program goes beyond what might be considered "traditional" technology transfer activities. Its main objectives are (1) to increase the awareness and level of information concerning transportation issues facing Region 2; (2) to improve the knowledge base and approach to problem solving of the region's transportation workforce, from those operating the systems to those at the most senior level of managing the system; and by doing so, to improve the overall professional capability of the transportation workforce; (3) to stimulate discussion and debate concerning the integration of new technologies into our culture, our work and our transportation systems; (4) to provide the more traditional but extremely important job of disseminating research and project reports, studies, analysis and use of tools to the education, research and practicing community both nationally and internationally; and (5) to provide unbiased information and testimony to decision-makers concerning regional transportation issues consistent with the UTRC theme.

### Project No(s):

UTRC/RF Grant No: 49997-34-24

**Project Date:** October 2014

**Project Title:** Real-time Dynamic Pricing for Bicycle Sharing Programs

### Project's Website:

<http://www.utrc2.org/research/projects/real-time-dynamic-pricing-bicycle-sharing-programs>

### Principal Investigator(s):

#### Changhyun Kwon

Assistant Professor

Department of Industrial and Systems Engineering  
318 Bell Hall

University at Buffalo (SUNY)

Buffalo, NY 14260

Tel: 716-645-4705

Email: [chkwon@buffalo.edu](mailto:chkwon@buffalo.edu)

### Co-authors:

#### Zulqarnain Haider

University at Buffalo (SUNY)

Buffalo, NY 14260

#### Alexander Nikolaev

Assistant Professor

Department of Industrial and Systems Engineering  
312 Bell Hall

University at Buffalo (SUNY)

Buffalo, NY 14260

Tel: (716) 645-4710

#### Jee Eun Kang

Assistant Professor

Department of Industrial and Systems Engineering  
409 Bell Hall

University at Buffalo (SUNY)

Buffalo, NY 14260

Tel: (716) 645-2357

Email: [jeeunka@buffalo.edu](mailto:jeeunka@buffalo.edu)

### Performing Organization:

University at Buffalo (SUNY)

To request a hard copy of our final reports, please send us an email at [utrc@utrc2.org](mailto:utrc@utrc2.org)

### Mailing Address:

University Transportation Research Center

The City College of New York

Marshak Hall, Suite 910

160 Convent Avenue

New York, NY 10031

Tel: 212-650-8051

Fax: 212-650-8374

Web: [www.utrc2.org](http://www.utrc2.org)

## Board of Directors

The UTRC Board of Directors consists of one or two members from each Consortium school (each school receives two votes regardless of the number of representatives on the board). The Center Director is an ex-officio member of the Board and The Center management team serves as staff to the Board.

### City University of New York

*Dr. Hongmian Gong - Geography/Hunter College*  
*Dr. Neville A. Parker - Civil Engineering/CCNY*

### Clarkson University

*Dr. Kerop D. Janoyan - Civil Engineering*

### Columbia University

*Dr. Raimondo Betti - Civil Engineering*  
*Dr. Elliott Sclar - Urban and Regional Planning*

### Cornell University

*Dr. Huaizhu (Oliver) Gao - Civil Engineering*

### Hofstra University

*Dr. Jean-Paul Rodrigue - Global Studies and Geography*

### Manhattan College

*Dr. Anirban De - Civil & Environmental Engineering*  
*Dr. Matthew Volovski - Civil & Environmental Engineering*

### New Jersey Institute of Technology

*Dr. Steven I-Jy Chien - Civil Engineering*  
*Dr. Joyoung Lee - Civil & Environmental Engineering*

### New York University

*Dr. Mitchell L. Moss - Urban Policy and Planning*  
*Dr. Rae Zimmerman - Planning and Public Administration*

### Polytechnic Institute of NYU

*Dr. Kaan Ozbay - Civil Engineering*  
*Dr. John C. Falcochio - Civil Engineering*  
*Dr. Elena Prassas - Civil Engineering*

### Rensselaer Polytechnic Institute

*Dr. José Holguín-Veras - Civil Engineering*  
*Dr. William "Al" Wallace - Systems Engineering*

### Rochester Institute of Technology

*Dr. James Winebrake - Science, Technology and Society/Public Policy*  
*Dr. J. Scott Hawker - Software Engineering*

### Rowan University

*Dr. Yusuf Mehta - Civil Engineering*  
*Dr. Beena Sukumaran - Civil Engineering*

### State University of New York

*Michael M. Fancher - Nanoscience*  
*Dr. Catherine T. Lawson - City & Regional Planning*  
*Dr. Adel W. Sadek - Transportation Systems Engineering*  
*Dr. Shmuel Yahalom - Economics*

### Stevens Institute of Technology

*Dr. Sophia Hassiotis - Civil Engineering*  
*Dr. Thomas H. Wakeman III - Civil Engineering*

### Syracuse University

*Dr. Riyad S. Aboutaha - Civil Engineering*  
*Dr. O. Sam Salem - Construction Engineering and Management*

### The College of New Jersey

*Dr. Thomas M. Brennan Jr - Civil Engineering*

### University of Puerto Rico - Mayagüez

*Dr. Ismael Pagán-Trinidad - Civil Engineering*  
*Dr. Didier M. Valdés-Díaz - Civil Engineering*

## UTRC Consortium Universities

The following universities/colleges are members of the UTRC consortium.

City University of New York (CUNY)  
Clarkson University (Clarkson)  
Columbia University (Columbia)  
Cornell University (Cornell)  
Hofstra University (Hofstra)  
Manhattan College (MC)  
New Jersey Institute of Technology (NJIT)  
New York Institute of Technology (NYIT)  
New York University (NYU)  
Rensselaer Polytechnic Institute (RPI)  
Rochester Institute of Technology (RIT)  
Rowan University (Rowan)  
State University of New York (SUNY)  
Stevens Institute of Technology (Stevens)  
Syracuse University (SU)  
The College of New Jersey (TCNJ)  
University of Puerto Rico - Mayagüez (UPRM)

## UTRC Key Staff

**Dr. Camille Kamga:** *Director, Assistant Professor of Civil Engineering*

**Dr. Robert E. Paaswell:** *Director Emeritus of UTRC and Distinguished Professor of Civil Engineering, The City College of New York*

**Herbert Levinson:** *UTRC Icon Mentor, Transportation Consultant and Professor Emeritus of Transportation*

**Dr. Ellen Thorson:** *Senior Research Fellow, University Transportation Research Center*

**Penny Eickemeyer:** *Associate Director for Research, UTRC*

**Dr. Alison Conway:** *Associate Director for Education*

**Nadia Aslam:** *Assistant Director for Technology Transfer*

**Nathalie Martinez:** *Research Associate/Budget Analyst*

**Tierra Fisher:** *Office Assistant*

**Bahman Moghimi:** *Research Assistant; Ph.D. Student, Transportation Program*

**Wei Hao:** *Research Fellow*

**Andriy Blagay:** *Graphic Intern*

1. Report No.		2. Government Accession No.		3. Recipient's Catalog No.	
4. Title and Subtitle <b>Real-time Dynamic Pricing for Bicycle Sharing Programs</b>				5. Report Date <b>October 15, 2014</b>	
				6. Performing Organization Code	
7. Author(s) <b>Zulqarnain Haider, Alexander Nikolaev, Jee Eun Kang, Changhyun Kwon</b>				8. Performing Organization Report No.	
9. Performing Organization Name and Address <b>University at Buffalo, the State University of New York Buffalo, NY 14260</b>				10. Work Unit No.	
				11. Contract or Grant No. <b>49997-34-24</b>	
12. Sponsoring Agency Name and Address <b>University Transportation Research Center 137th Street and Convent Avenue New York, NY 10031</b>				13. Type of Report and Period Covered <b>Final Report</b>	
				14. Sponsoring Agency Code	
15. Supplementary Notes					
16. Abstract <p>This paper presents a new conceptual approach to improve the operational performance of public bike sharing systems using pricing schemes. Its methodological developments are accompanied by experimental analyses with bike demand data from Capital Bikeshare program of Washington, DC. An optimized price vector determines the incentive levels that can persuade system customers to take bicycles from, or park them at, neighboring stations so as to strategically minimize the number of unbalanced stations. This strategy intentionally makes some unbalanced stations even more highly unbalanced, creating hub stations. This reduces the need for trucks and dedicated staff to carry out inventory repositioning. For smaller networks, a bilevel optimization model is introduced to minimize the number of unbalanced stations optimally. The results are compared with two heuristic approaches. One approach involves a genetic algorithm, while the second adjusts route prices by segregating the stations into different categories based on their current inventory profile, projected future demand, and maximum and minimum inventory values calculated to fulfill certain desired service level requirements. It is shown that the latter approach, called the iterative price adjustment scheme (IPAS), reduces the overall operating cost while partially or fully obviating the need for a manual repositioning operation.</p>					
17. Key Words <b>bike-sharing; shared-mobility; rebalancing; pricing; heuristics</b>			18. Distribution Statement		
19. Security Classif. (of this report) <b>Unclassified</b>		20. Security Classif. (of this page) <b>Unclassified</b>		21. No of Pages <b>29</b>	22. Price

**Disclaimer**

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. The contents do not necessarily reflect the official views or policies of the UTRC, the Department of Transportation or the Federal Highway Administration. This report does not constitute a standard, specification or regulation. This document is disseminated under the sponsorship of the Department of Transportation, University Transportation Centers Program, in the interest of information exchange. The U.S. Government and the Department of Transportation assume no liability for the contents or use thereof.

# 1 Introduction

Increasingly, Public Bike Sharing Systems (BSS) are being adopted by many major cities throughout the world. Bicycles are being touted as a way to achieve sustainable mobility in an urban setting while also helping to alleviate the last mile problem in urban transportation (Shaheen et al., 2010). As of September 19, 2014, some form of bike sharing system is operating in 747 cities worldwide and another 235 such systems are in planning or under construction with more and more cities becoming interested (Meddin and DeMaio, 2014). One of the major problems facing these systems is the operational issue of repositioning of bicycles between different stations. Due to demand variability, certain stations become too full or too empty to effectively service new customers. This not only affects the desired service level but also incurs spurious operational costs. According to a report by New York City Department of City Planning (2009) based on different case studies, the total capital costs for a bike sharing system vary from \$3000/bike to \$4400/bike in different cities. When averaged across programs, the average yearly operating cost for a bike share program is around \$1,600/bicycle.

The operating cost consists mainly of system operations, administration, marketing and utility costs associated with hardwired stations. System operation forms the largest share of these costs and will include functions such as: maintenance of all equipment, rebalancing of bicycles, customer service operations, and website and IT support (The Pennsylvania Environmental Council, 2013). Clearly, the repositioning of bicycles from stations too full to stations too empty is a big operational overhead. In fact, for Vélib system in Paris, the average cost of a single repositioning for a single bike is \$3 (DeMaio, 2009). A system-wide snapshot of Capital Bikeshare at 9:30 AM on May 15, 2014 shows that 88 out of 202 stations are *unbalanced* considering 90% service level (see Section 3.1).

This paper aims to reduce the number of such unbalanced stations by rebalancing their inventory through price incentives/disincentives. To do so, we will intentionally make some unbalanced stations even more unbalanced, making these unbalanced stations function as *hubs*. If only a few highly unbalanced stations exist in the system, then bike redistribution can be handled with just a few regular short time truck trips, in every operational period (e.g., a 30-minute-long one).

With the reduced number of unbalanced stations, the operation of truck redistribution becomes simpler, more efficient and thus results in operating cost reduction. This observation is key to our idea of designing dynamic pricing policies. We seek to ensure that surplus bicycles are gathered predominantly at designated ‘surplus accumulation’ stations, and similarly, the deficiency/lack of bicycles predominantly occurs at ‘lack accumulation’ stations.

We find that we can successfully reduce the number of these unbalanced stations, by giving travelers multiple journey choices and by changing the cost of those journeys through pricing. We also show that the cost of the same degree of manual rebalancing outweighs the price incentives offered. To determine such price incentives, we first formulate a bi-level optimization model and provide a single-level reformulation that may be useful for small networks. We propose a heuristic algorithm, called the iterative price adjust scheme (IPAS), and compare its performance with the single-level optimization model solved by a commercial solver and solutions obtained by a genetic algorithm.

The performance of IPAS is demonstrated by computational experiments. Using the data from Capital Bikeshare in Washington, D.C., we show how our approaches manage to successfully minimize the number of unbalanced stations. The efficacy of our heuristic approaches vis-à-vis execution time while bringing satisfactory improvement to the overall objective of minimizing the number of unbalanced stations is also shown.

## 2 Literature Review

Bike sharing systems have recently garnered an increased interest from the research community due to their increased importance in sustainable urban transport systems. DeMaio (2009) and Shaheen et al. (2010) separately discuss the history, the impacts, the models of provision and the future of public BSS. Introducing what they call the Fourth generation of BSS, they identify improved redistribution of bicycles as a key challenge facing BSS. Schuijbroek et al. (2013) have an excellent and comprehensive description of BSS literature. They divide up the BSS literature into four major streams including strategic design, demand analysis, service level analysis and rebalancing operations. We thus refer readers to Schuijbroek et al. (2013) for general literature review, and limit

this Section to reviewing most relevant literature on bike sharing systems, in particular, rebalancing operations.

Rebalancing operations are a big part of operating costs of a bike sharing system (The Pennsylvania Environmental Council, 2013). Most such systems are run in collaboration with city governments and depend financially on public funds and corporate sponsorship. Because these systems are often cash strapped, it is not possible to indiscriminately add extra bicycles or docks to the system. Furthermore, the demand also depends on many factors and is hardly predictable. This necessitates some kind of rebalancing of the system. Generally, bike sharing systems employ two methods to redistribute the bicycles: truck-based manual redistribution and pricing-based rebalancing.

Most bike sharing systems have a fleet of trucks that move around and pick up and drop bicycles. Vélib has 20 trucks (Benchimol et al., 2011) operating 24 hours to carry out manual rebalancing. Trucks and crew required to operate them have huge associated costs. Paul DeMaio of MetroBike, LLC, mentions a conversation with Luud Schimmelpennick, a pioneer of bike sharing concept, in DeMaio (2009). He reports that according to Schimmelpennick the cost for distribution of a single bike for JCDecaux is \$3 and that any scheme that offers incentives to customers would increase the redistribution efficiency at a fraction of the current cost. Since some kind of manual balancing is always required, most of rebalancing literature is focused on optimal truck routing. Benchimol et al. (2011) introduce several approximation algorithms for static rebalancing of bicycles at the end of the day when there is not much bike movement happening.

Several papers have recently studied truck-based manual bicycle redistribution. Raviv et al. (2013) introduce several formulations for static rebalancing problem but their objective function minimizes the expected user dissatisfaction rather than minimizing the total travel distance. Fricker and Gast (2014) study a simple model with symmetry where all the bicycle stations have the same parameters. The authors establish that even in this perfect scenario, the probability of a station being full or empty is  $2/(K + 1)$  where  $K$  is the capacity of each station, and then, propose to improve this situation through incentives and regulation. Contardo et al. (2012) introduce a dynamic public bike sharing balancing problem (DPBSBP) to rebalance a BSS during daytime

peak hours. They solve the DPBSBP problem using Dantzig-Wolfe decomposition and Benders decomposition to derive lower bounds and fast feasible solutions. Caggiani and Ottomanelli (2012) construct a modular Decision Support System (DSS) for dynamic bike redistribution. Shu et al. (2013) discuss under-utilization of bike sharing systems in Chinese cities and propose a deterministic model to optimally deploy bicycles and docking capacity at different stations. They also evaluate the value of redistribution and its impact on the number of trips supported by the system. They conclude that for systems with more than 30,000 bicycles, frequent periodic redistribution does not add much value and a small number of daily redistributions are recommended.

There is a recent trend in BBS literature to introduce a scheme of incentives to get users to move bicycles away from the stations too full and into the stations too empty. Vélib operates a V+ scheme to induce users to avoid certain stations and prefer others. Users get 15 minutes of added travel time if they place the bicycles at one of the hundred uphill stations (Fricker and Gast, 2014). The incentives can be in the form of extra added time, as is the case with Vélib, or some cash discounts. The literature on user incentive schemes is not as plentiful as that on rebalancing through trucks (Fricker and Gast, 2014). Fricker and Gast (2014) present a two-choice model in which each user is provided with two station choices at the time of a rental and is given an incentive to choose the station with the lower load as a destination. They show that even if a fraction of the users make the intended choice, the number of unbalanced stations comes down dramatically. Wasserhole et al. (2012) also develop a pricing strategy using fluid approximation.

Pfrommer et al. (2014) introduce a tailored algorithm for dynamic route planning for multiple trucks for redistribution of bicycles and then go on to devise a system of price incentives computed based on Model Predictive Control (MPC) to draw users away from full or empty stations. They attempt to increase the service level of the system by adding repositioning trucks and increasing the incentive payouts. They observe that with the increase in number of trucks and incentive payouts, diminishing gains to service level are reported. They also conclude that a system of price incentives is more effective on weekends as compared to weekdays when work related commuting takes place.

We aim to establish a system of cash discounts and penalties on user fee to modify user decisions. The operators have real time status data on all stations and based on this data a price vector can

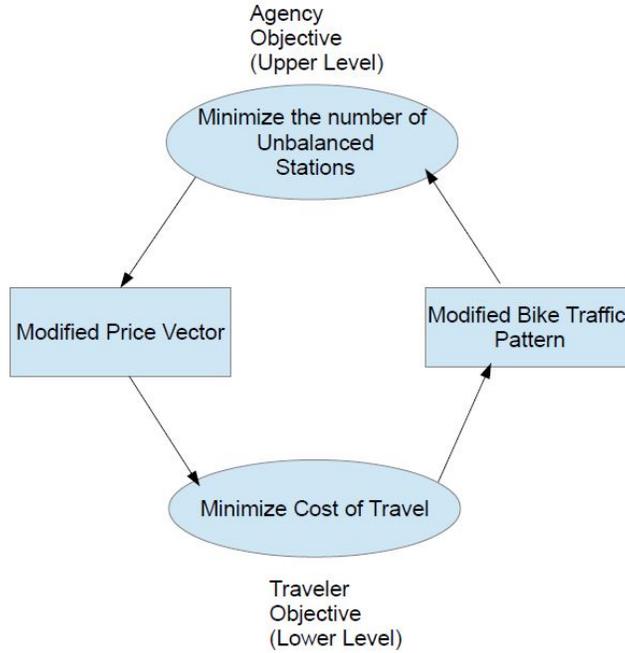


Figure 1: Bilevel Decision Making Process

be calculated for all journeys. This information can be provided to users at Point of Sale or through mobile apps. As described by Waserhole et al. (2012), price incentives can be offered in discrete jumps with certain small increments, or they can be continuous within a range with a certain maximum and minimum value. Pricing policies can also be dynamically changed in real-time depending upon the current state of the system and the expected future demand, or they can be static, i.e., independent of the system’s current state, and set in advance.

### 3 Model Formulation

The decision-making process for bike sharing systems is bilevel, as shown in Figure 1. Decisions regarding the location and size of stations as well as pricing are made by the operator running the system, while lower level journey decisions are made by the customers using the bicycles. In our model, the upper level (operator) objective is to minimize the total number of *unbalanced* stations. The lower level customer objective is to make a journey between two points at the minimum possible cost. The underlying assumption is that travelers will always take the minimum cost route. This section works to develop a detailed formulation of the problem that exploits the idea of strategic

customer incentives. A few explanations, leading to the definition of *unbalanced* stations, are first in order.

### 3.1 Service Level Requirements

As described earlier, bike sharing systems are subject to two demands. On one hand, there exists the demand for bicycles while on the other hand, there is the demand for empty docks, i.e., for parking the bicycles at the end of the journey. A bike procured from one station is eventually parked back at the same station or any other station in the system. Every time a station is full or empty, a service opportunity is wasted. Most bike sharing systems keep track of stations being empty or full. Some systems measure the number of instances (e.g. Capital Bike share in Washington, DC) while others measure the fraction of time (e.g. Vélib in Paris) that the stations are empty or full. Operators do it for more efficient rebalancing operations and to determine the need for expansion or reduction in the number of docks available at a particular station. Schuijbroek et al. (2013) define a measurable Type-2 service level: the fraction of demand satisfied directly should be larger than  $\beta_h^-$  for pickups and larger than  $\beta_h^+$  for returns, assuming no back-orders. We use the same definition:

$$\frac{E[\text{Satisfied bike pickup demands}]}{E[\text{Total bike pickup demands}]} \geq \beta_h^-,$$

$$\frac{E[\text{Satisfied bike return demands}]}{E[\text{Total bike return demands}]} \geq \beta_h^+.$$

Schuijbroek et al. (2013) then go on to establish a method to evaluate the values of minimum and maximum inventory for each station in the system, respectively designated as  $I_h^{\min}$  and  $I_h^{\max}$  for given  $\beta_h^-$  and  $\beta_h^+$ . They model the inventory  $I_h$  at station  $h$  as an  $M/M/1/K$  queuing process with customers in the queue for bicycles or docks representing the inventory. We use their system of equations to evaluate the values of  $I_h^{\min}$  and  $I_h^{\max}$ . For given  $\beta_h^-$  and  $\beta_h^+$ , starting inventory  $I_h^0$  should ideally be rebalanced so that:

$$I_h^{\min} \leq I_h \leq I_h^{\max}.$$

If a station does not satisfy the above inequality, the station is termed *unbalanced*. It bears repeating that a station is lack-unbalanced when  $I_h \leq I_h^{\min}$  and surplus-unbalanced when  $I_h \geq I_h^{\max}$ . It must be noted that  $I_h^{\min}$  is always greater than or equal to 0 and  $I_h^{\max}$  is always less than or equal to the maximum capacity of station  $h$ , i.e., the number of docks installed in station  $h$ . So a station can be unbalanced even when it is not completely full or empty. The following parameters are required to calculate  $I_h^{\min}$  and  $I_h^{\max}$  for station  $h$ :  $\beta_h^-$ ,  $\beta_h^+$ , number of docks installed, the rate of arrival of users to pick up bicycles, and the rate of arrival of users to return bicycles.

### 3.2 A Bi-Level Formulation

This section presents the mathematical formulation of the bilevel problem. The first level represents the price change vector to minimize the number of *unbalanced* stations while the lower level corresponds to a minimum cost network flow problem which determines the route choices made by the travelers. Let  $\mathcal{S}$  be the set of stations in the system. Let us assume that for a single journey,  $r$  is the origin station and  $s$  is the destination station where  $r, s \in \mathcal{S}$ . Let us also assume that  $(r, s)$  is the OD pair for a single one way trip and  $\mathcal{W}$  is the set of all possible OD pairs, i.e.,  $(r, s) \in \mathcal{W}$ . If the number of stations in the network is  $n$  then the number of OD pairs is  $n^2$ . The distance  $d$  between two stations is the distance along the shortest bike route and not the euclidean distance. In an urban setting, each station has a number of *neighborhood* stations. We assume that two stations less than 600 meters apart are *neighborhood* stations. On average, this accounts for less than 6 minutes of walking. For every OD pair, both origin and destination have a number of neighborhood stations as illustrated in Figure 2. The colored circles around the stations represent the neighborhood radius.

Let us designate a full, directed network  $G(\mathcal{N}, \mathcal{A}, \mathcal{P})$  where  $\mathcal{N}$  denotes the set of nodes in the network,  $\mathcal{A}$  denotes the set of arcs and  $\mathcal{P}$  denotes the set of paths. In this network a direct arc between any two nodes is also the shortest path between them. For every single OD pair in set  $\mathcal{W}$ , we construct a smaller directed network  $G^{rs}(\mathcal{N}^{rs}, \mathcal{A}^{rs}, \mathcal{P}^{rs})$ . Observe that  $\mathcal{N}^{rs} \subset \mathcal{N}$  but  $\mathcal{A}^{rs} \not\subset \mathcal{A}$  and  $\mathcal{P}^{rs} \not\subset \mathcal{P}$ . As shown in Figure 2, a traveler intending to go from origin  $r$  to destination  $s$  can now take any one of the many paths available to him. If  $No_r$  and  $No_s$  are the numbers of

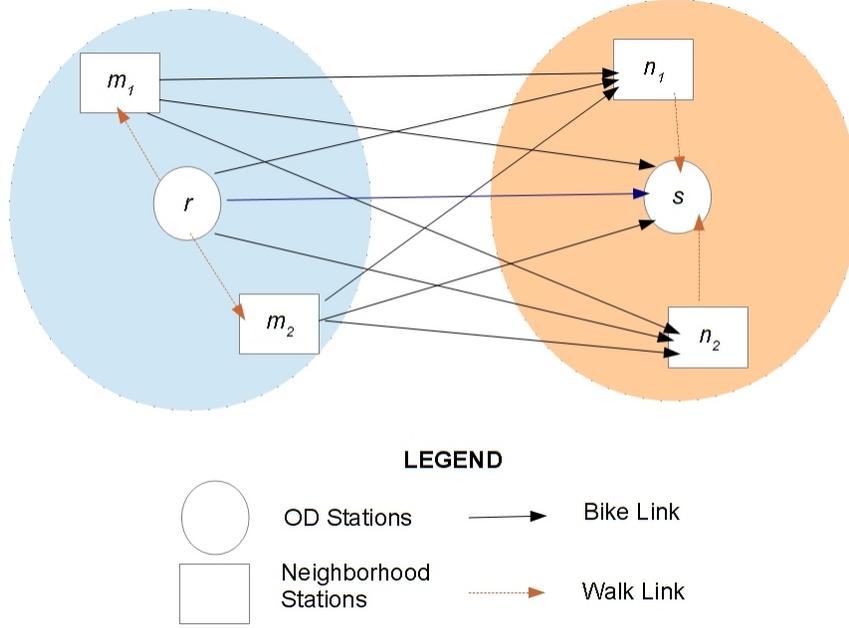


Figure 2: Network structure for every OD station pair

neighborhood points of origin and destination, respectively, then the total number of alternate paths available is  $(No_r + 1) \times (No_s + 1)$ . Each path can consist of one, two or three links. For example, path  $r \rightarrow s$  consists of one link, while path  $r \rightarrow m_2 \rightarrow n_1 \rightarrow s$  consists of three links. A traveler taking the latter path walks from origin  $r$  to its neighborhood point  $m_2$ , rents a bicycle and bikes to destination's neighborhood point  $n_1$ , then parks the bicycle at an empty dock and walks to the destination  $s$ . In Figure 2, *bike links* are represented by solid arrows, while dotted arrows represent *walk links*. Note that every path contains one and only one bike link.

For a given OD pair  $(r, s)$ , let  $C_{mn}^{rs}$  be the cost matrix for all links  $(m, n)$  where  $(m, n) \in \mathcal{A}^{rs}$  and  $m, n \in \mathcal{S}$ . The cost  $C_{mn}^{rs}$  of traversing a link consists of various subcosts. These include the cost of walking, cost of biking, and the price of renting a bicycle to travel on a bike link. The rental price is determined by the operator. Let these subcosts be denoted by  $w_{mn}^{rs}, v_{mn}^{rs}$  and  $p_{mn}$ , respectively. The costs  $w_{mn}^{rs}, v_{mn}^{rs}$  are calculated as

$$w_{mn}^{rs} = \nu_3 t_{mn}^{rs} \quad \text{and} \quad v_{mn}^{rs} = \nu_2 t_{mn}^{rs}.$$

In the above expressions  $t_{mn}^{rs}$  is the time of travel between two stations  $m$  and  $n$  using a bicycle.

$\nu_2$  and  $\nu_3$  are the coefficients that convert distance between stations to the cost depending upon biking and walking travel times, respectively. As stated earlier, the price  $p_{mn}$  is defined by the operator. The price associated with a single given link  $(m, n)$  in multiple OD pair networks is the same. For example, the value of  $p_{19,3}$  is the same for all  $(19, 3)$  links in all the feasible OD pair networks. Hence, all  $p_{mn}$  values in an OD pair network form a price vector associated with the OD pair  $(m, n)$  and defined by the operator.

Also, for every OD pair  $(r, s)$ , let  $(m, n) \in \mathcal{A}_B^{rs}$  be the links where a bicycle is used to traverse and hence the cost of travel consists of  $p_{mn}$  and  $v_{mn}^{rs}$ . Similarly, Let  $(m, n) \in \mathcal{A}_W^{rs}$  be the links where a traveler walks and hence the cost of travel consists of  $w_{mn}^{rs}$  alone. The price vector  $p_{mn}$  further consists of a fixed component and a variable component,

$$p_{mn} = p_{mn}^0 + q_{mn}, \quad (1)$$

where  $p_{mn}^0$  is the fixed base price set by the operator and depending on the time of the journey  $(m, n)$  while  $q_{mn}$  is the variable component capturing a penalty or incentive within a fixed range  $[q_{mn}^{\min}, q_{mn}^{\max}]$ , determined again by the operator, with

$$q_{mn}^{\min} \leq q_{mn} \leq q_{mn}^{\max}. \quad (2)$$

Note that all the costs mentioned above except  $q_{mn}$  are fixed costs depending only on travel time. By modifying the price change vector  $q_{mn}$ , the operator can modify the cost matrix and influence the traveler's decision about which path to take. Here we introduce a binary variable  $x_{mn}^{rs}$  which is equal to 1 if link  $(m, n)$  is used to travel between OD pair  $(r, s)$ , and 0 otherwise. Since for every OD pair, one or more links  $(m, n)$  can be used to travel between origin  $r$  and destination  $s$ , one or more variables  $x_{mn}^{rs}$  can take on the value of 1. The outcome  $x_{mn}^{rs}$  of the lower level program which depends on traveler choices is used at the upper level to calculate the bicycle inventory  $I_h$  for each station  $h$  at each instant. This information feeds into the upper level objective function to minimize the number of unbalanced stations. To calculate the inventory, we only require links where a bicycle is used. Let parametric vector  $\delta_{mn}^{rs}$  have value 1 for a bike link and 0 for a walk

Table 1: Mathematical Notation

$\mathcal{S}$	Set of stations
$\mathcal{W}$	Set of Origin-Destination (OD) pairs
$\mathcal{A}$	Set of arcs in the directed network for every OD pair
$\mathcal{P}$	Set of Possible Paths for each OD pair
$I_h^{\max}$	Maximum number of permissible bicycles at a station $h$ , beyond which the station is considered unbalanced
$I_h^{\min}$	Minimum number of permissible bicycles at a station $h$ , beyond which the station is considered unbalanced
$I_h^0$	Starting level of bike inventory at station $h$ at any time $t$
$I_h$	Current level of bike inventory at station $h$ at a later time $t + \delta t$
$\delta_{mn}^{rs}$	1 for a bike link and 0 for a walking link
$\alpha_{mn}^{rs}$	1 for a walking link and 0 for a bike link
$y_h$	1 for a surplus station where surplus station is a station where $I_h > I_h^{\max}$
$z_h$	1 for a lack station where lack station is a station where $I_h < I_h^{\min}$
$q_{mn}$	Price change (Incentive or penalty) for traversing link $(m, n)$
$x_{mn}^{rs}$	1 when link $(m, n)$ is used to travel between OD pair $(r, s)$
$d^{rs}$	Demand for OD pair $(r, s)$ in current time period
$\text{No}_r$	The number of neighborhood points of origin
$\text{No}_s$	The number of neighborhood points of destination

link. Then the product  $\delta_{mn}^{rs} x_{mn}^{rs}$  is 1 only if a link  $(m, n)$  used to travel between an OD pair is a bike link and 0 if it is not a bike link. Observe that for every OD pair combination,  $\delta_{mn}^{rs} x_{mn}^{rs}$  is 1 for only one link  $(m, n)$ . Table 1 details the mathematical notation used in the model.

### The Upper Level Pricing Problem of the System Operator

Using the notation defined in Table 1, we formulate an upper level optimization problem to determine the price change vector  $q_{mn}$  as follows:

$$(P) \quad \min_{x,y,z,p} \quad \sum_h y_h + \sum_h z_h \tag{3}$$

subject to

$$I_h - I_h^{\max} \leq M y_h \quad \forall h \in \mathcal{S}, \tag{4}$$

$$I_h^{\min} - I_h \leq M z_h \quad \forall h \in \mathcal{S}, \tag{5}$$

$$I_h = I_h^0 - \sum_{(r,s) \in \mathcal{W}} \sum_{n \in \mathcal{N}^{rs}} x_{hn}^{rs} \delta_{hn}^{rs} d^{rs} + \sum_{(r,s) \in \mathcal{W}} \sum_{n \in \mathcal{N}^{rs}} x_{nh}^{rs} \delta_{nh}^{rs} d^{rs} \quad \forall h \in \mathcal{S}, \quad (6)$$

$$q_{mn}^{\min} \leq q_{mn} \leq q_{mn}^{\max} \quad \forall (m, n) \in \mathcal{A}, \quad (7)$$

where values of variables  $x_{mn}^{rs}$  depend on the minimum cost route choice of bike users and is determined by a lower level problem to be introduced later.

### The Lower Level Routing Problem of the Bike Users

In the lower level problem bike users who want to travel from origin  $r$  to destination  $s$  use minimum cost paths so the objective is,

$$\min \sum_{(r,s) \in \mathcal{W}} \sum_{(m,n) \in \mathcal{A}^{rs}} c_{mn}^{rs} x_{mn}^{rs} \quad (8)$$

subject to

$$\sum_{(m,n) \in \mathcal{A}^{rs}} x_{mn}^{rs} - \sum_{(n,m) \in \mathcal{A}^{rs}} x_{nm}^{rs} = e_m^{rs} \quad \forall m \in \mathcal{N}^{rs}, (r, s) \in \mathcal{W}, \quad (9)$$

$$x_{mn}^{rs} \in \{0, 1\} \quad \forall (m, n) \in \mathcal{A}^{rs}, (r, s) \in \mathcal{W}, \quad (10)$$

where

$$c_{mn}^{rs} = \begin{cases} p_{mn}^0 + q_{mn} + \nu_2 v_{mn}^{rs} & \forall (m, n) \in \mathcal{A}_B^{rs} \\ \nu_3 w_{mn}^{rs} & \forall (m, n) \in \mathcal{A}_W^{rs}. \end{cases} \quad (11)$$

In (9),  $e_m^{rs}$  takes the value 1 (respectively, -1), if node  $m$  is the origin (respectively, destination) of the trip, and 0 otherwise.

### 3.3 A Single Level Reformulation

In the lower level problem the integrality requirement for variables  $x_{mn}^{rs}$  can be replaced by the constraints  $x_{mn}^{rs} \geq 0$ . This is so because the lower level program is a minimum cost network flow problem: its right hand side can only be integer and the coefficient matrix in the left hand side forms a unimodular matrix. Now we can represent the lower problem by its optimality conditions

or KKT conditions of its LP relaxation. Since the lower-level problem is a linear optimization problem, we can replace it by following Karush-Kuhn-Tucker (KKT) optimality conditions:

$$\delta_{mn}^{rs}(p_{mn}^0 + q_{mn} + \nu_2 v_{mn}^{rs}) + \nu_3 \alpha_{mn}^{rs} w_{mn}^{rs} - \lambda_m^{rs} + \lambda_n^{rs} - \mu_{mn}^{rs} = 0 \quad \forall (m, n) \in \mathcal{A}^{rs}, (r, s) \in \mathcal{W}, \quad (12)$$

$$\sum_{(m,n) \in \mathcal{A}^{rs}} x_{mn}^{rs} - \sum_{(n,m) \in \mathcal{A}^{rs}} x_{nm}^{rs} = e_m^{rs} \quad \forall m \in \mathcal{N}^{rs}, (r, s) \in \mathcal{W}, \quad (13)$$

$$-x_{mn}^{rs} \leq 0 \quad \forall (m, n) \in \mathcal{A}^{rs}, (r, s) \in \mathcal{W}, \quad (14)$$

$$\mu_{mn}^{rs} \geq 0 \quad \forall (m, n) \in \mathcal{A}^{rs}, (r, s) \in \mathcal{W}, \quad (15)$$

$$\lambda_m^{rs} \text{ free} \quad \forall (m, n) \in \mathcal{A}^{rs}, (r, s) \in \mathcal{W}, \quad (16)$$

$$-\mu_{mn}^{rs} x_{mn}^{rs} = 0 \quad \forall (m, n) \in \mathcal{A}^{rs}, (r, s) \in \mathcal{W}. \quad (17)$$

The complementary slackness conditions (17) are non-convex, and we can linearize them taking advantage of binary nature of  $x_{mn}^{rs}$ . Suppose  $M$  is a very large number then the linearized constraint would be:

$$\mu_{mn}^{rs} \leq M(1 - x_{mn}^{rs}) \quad \forall (m, n) \in \mathcal{A}^{rs}, (r, s) \in \mathcal{W}. \quad (18)$$

We state the final formulation of our optimization model here:

$$(P) \quad \min_{x,y,z,p} \sum_h y_h + \sum_h z_h \quad (19)$$

subject to

$$I_h - I_h^{\max} \leq M y_h \quad \forall h \in \mathcal{S}, \quad (20)$$

$$I_h^{\min} - I_h \leq M z_h \quad \forall h \in \mathcal{S}, \quad (21)$$

$$I_h = I_h^0 - \sum_{(r,s) \in \mathcal{W}} \sum_{n \in \mathcal{N}^{rs}} x_{hn}^{rs} \delta_{hn}^{rs} d^{rs} + \sum_{(r,s) \in \mathcal{W}} \sum_{n \in \mathcal{N}^{rs}} x_{nh}^{rs} \delta_{nh}^{rs} d^{rs} \quad \forall h \in \mathcal{S}, \quad (22)$$

$$q_{mn}^{\min} \leq q_{mn} \leq q_{mn}^{\max} \quad \forall (m, n) \in \mathcal{A}, \quad (23)$$

$$\delta_{mn}^{rs}(p_{mn}^0 + q_{mn} + \nu_2 v_{mn}^{rs}) + \nu_3 \alpha_{mn}^{rs} w_{mn}^{rs} - \lambda_m^{rs} + \lambda_n^{rs} - \mu_{mn}^{rs} = 0 \quad \forall (m, n) \in \mathcal{A}^{rs}, (r, s) \in \mathcal{W}, \quad (24)$$

$$\sum_{mn \in \mathcal{A}^{rs}} x_{mn}^{rs} - \sum_{nm \in \mathcal{A}^{rs}} x_{nm}^{rs} = e_m^{rs}, \quad \forall m \in \mathcal{N}^{rs}, (r, s) \in \mathcal{W}, \quad (25)$$

$$-x_{mn}^{rs} \leq 0 \quad \forall (m, n) \in \mathcal{A}^{rs}, (r, s) \in \mathcal{W}, \quad (26)$$

$$\mu_{mn}^{rs} \geq 0 \quad \forall (m, n) \in \mathcal{A}^{rs}, (r, s) \in \mathcal{W}, \quad (27)$$

$$\lambda_m^{rs} \text{ free} \quad \forall (m, n) \in \mathcal{A}^{rs}, (r, s) \in \mathcal{W}, \quad (28)$$

$$\mu_{mn}^{rs} \leq M(1 - x_{mn}^{rs}) \quad \forall (m, n) \in \mathcal{A}^{rs}, (r, s) \in \mathcal{W}. \quad (29)$$

In the final model represented by Equations (19) to (29), for every OD pair  $(r, s)$ , the number of decision variables is calculated as  $2\text{No}_r\text{No}_s + 3(\text{No}_r + \text{No}_s)$ . For a network of 200 stations, this amounts to approximately 1 million variables and 1 million constraints. Therefore, while this formulation can be useful for a comparative evaluation of other methods (with small-scale problems), more scalable solutions are desirable for practical purposes.

## 4 Computational Methods

This section presents two heuristic methods to solve model (P). Keeping the upper level problem intact, we work with the lower level problem: instead of performing the exact minimum cost optimization to determine the optimum price change vector  $q_{mn}$ , we determine  $q_{mn}$  heuristically. The overall objective of minimizing the number of unbalanced stations remains the same. Section 4.1 presents a genetic algorithm based heuristic to produce the price change vector (Details in Appendix A). Another heuristic method, IPAS, presented in Section 4.2 produces a price change vector using discrete increments and decrements on the price between different station categories.

### 4.1 Continuous Genetic Algorithm

Genetic Algorithms (GA) are evolutionary algorithms that use natural selection for generating solutions to optimization problems. We use continuous form of GA to determine price change vector  $q_{mn}$ . The variable  $q_{mn}$  is continuous within a pre-specified range. In continuous GA, variables are

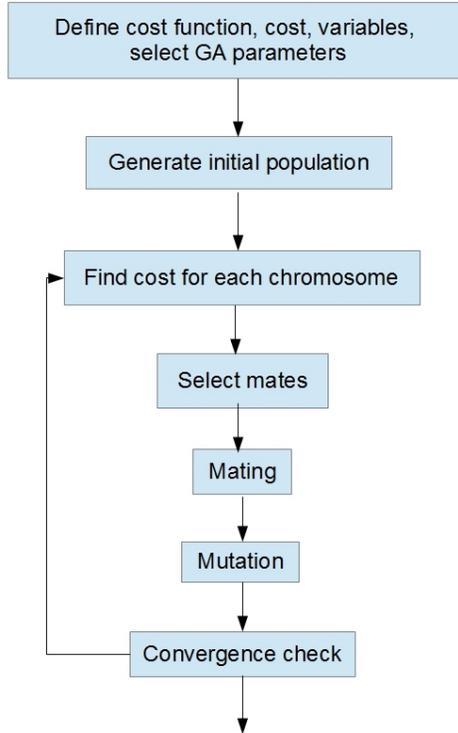


Figure 3: Flow chart of a continuous GA

represented as floating point numbers over a particular range. Continuous GA has many advantages over binary GA (Haupt and Haupt, 2004). Figure 3, taken from Haupt and Haupt (2004), shows major steps in a continuous GA. Although these steps are roughly the same for binary GA as well, the difference lies in the use of crossover and mutation operators. The details of the continuous GA applied to model (P) are provided in Appendix A.

## 4.2 Iterative Price Adjustment Scheme

We present the iterative price adjustment scheme (IPAS), which relies on a simple decision making process that classifies the bike stations into different categories based on their starting inventory level  $I_h^0$ , the maximum and minimum inventory values, and the demand in the next time period. Based on these factors we divide the stations into six different types: 1) Balanced Stations with Bikes Needed, 2) Balanced Stations with Docks Needed, 3) Unbalanced Stations with Bikes Needed, 4) Unbalanced Stations with Docks Needed, 5) HUB Stations with Bikes Needed (also called Lack accumulation stations), and 6) HUB Stations with Docks Needed (also called Surplus accumulation

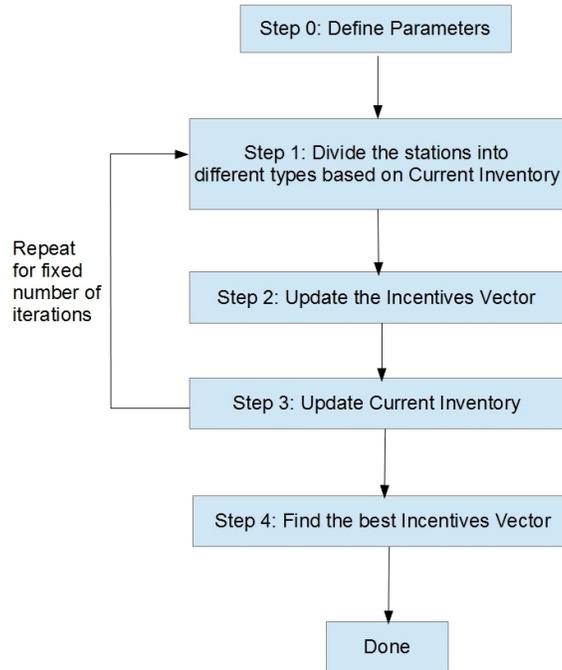


Figure 4: Flow chart of the Iterative Price Adjustment Scheme (IPAS)

stations).

We will identify stations that are “slightly” unbalanced at the current time and try to make those stations balanced. We will also identify stations that are “highly” unbalanced and try to make them even more unbalanced for the sake of preserving the balance at other neighboring stations.

As the primary objective of the pricing problem lies in identifying the accumulation stations, while simultaneously reducing the number of such stations, we will first develop a heuristic to identify such stations where accumulation happens naturally. Further accumulation may be possible and likely whenever a station is expected to experience an increase in the imbalance at the current pricing scheme. Here, we will look at the possibility of transferring the surplus inventory from high surplus stations to the neighboring stations that lack bicycles. If a station is expected to have a significant surplus and there are no stations in its neighborhood where lack accumulation is expected, then such a station is a good candidate for a surplus accumulation hub. For such a station, to help further accumulation, we can decrease prices for bike returns to that station and increase regular prices on bike checkouts from that station. For a likely lack accumulation station, we can do the opposite: increase regular prices on bike returns and offer discounts for checkouts.

Based on such selection of accumulation stations and price changes, we may evaluate the objective function in (P) and iteratively adjust prices. Figure 4 shows a flow chart for the proposed iterative price adjustment scheme.

#### 4.2.1 Step 0: Define the Algorithm Parameters

First of all, we define some parameters that will be used in the proposed scheme. The basic parameters are as follows: The parameters  $I_h^{\max}$ ,  $I_h^{\min}$ ,  $I_h^0$ , and  $I_h$  as defined in Table 1;  $k$  is an integer number representing the running iteration of the algorithm; and  $q_{mn}^k$  is the Price change (Incentive or penalty) vector for traversing link  $(m, n)$  during iteration number  $k$  of the algorithm where  $q_{mn}$  for a given link  $(m, n)$  is the same for all OD pairs. The starting price change vector called  $q_{mn}^k$  for  $k = 0$  can generally have all values equal to 0.

Parameter  $\theta_h^{\text{in}}$  describes the maximum number of bicycles that can possibly come into a certain station  $h$ , which is the sum of the number of bicycles coming in from all the other stations of the network into the station  $h$  and the number of bicycles coming in from all the other stations of the network into the neighborhood stations of  $h$ . We assume that if big enough incentives were offered, all the traffic coming into neighborhood points of  $h$  will be redirected to  $h$  and users will take rest of the trip walking.

Parameter  $\theta_h^{\text{out}}$  describes the maximum number of bicycles that can possibly go out of a certain station  $h$ , which is the sum of the number of bicycles going to all the other stations of the network from the station  $h$  and the number of bicycles going to all the other stations of the network from the neighborhood stations of  $h$ . We assume that if big enough incentives were offered, all the traffic going from neighborhood points of  $h$  will be redirected through  $h$  and users will walk to  $h$  and take a bike onward to their destinations.

The *rank ratio*, denoted by  $\rho$ , is the parameter used to rank different stations based on their current inventory status,

$$\rho_h = \frac{(I_h - I_h^{\max})^2 + (I_h - I_h^{\min})^2}{(I_h^{\max} - I_h^{\min})^2}. \quad (30)$$

The value of  $\rho$  varies from 0.5 to  $\infty$  with lesser values implying a more balanced station and larger

values implying a more unbalanced station. A station with value of  $\rho \geq 1$  implies that either of the two inequalities  $I_h \leq I_h^{\max}$  and  $I_h \geq I_h^{\min}$  are unsatisfied. The rank ratio is used to sort stations that fall into one of the six categories listed below.

#### 4.2.2 Step 1: Define Station Types

The stations fall in only one of the six subsets of stations based on the following definitions:

**HUB Stations with Bikes Needed (HSBN)** have very low bike inventory, so that we would not have enough bicycles to make this station balanced again, even if the incoming bike inventory from all the neighboring stations were to be redirected; hence they become good candidates for Lack Accumulation Stations. Instead of making such a station balanced by gaining bicycles, we make it even more unbalanced by losing bicycles further. Such a station must satisfy the following inequality:

$$\pi I_h^{\min} - I_h^0 \leq \theta_h^{\text{in}}.$$

**HUB Stations with Docks Needed (HSDN)** have very high bike inventory so that it is very difficult to lose enough bicycles to make the station balanced again, even if all the outgoing bicycles from all its neighborhood stations were to be redirected through it. Hence these stations become good candidates for Surplus Accumulation Station. So instead of making it balanced by losing bicycles, we make it even more unbalanced by gaining bicycles further. Such station must satisfy the following inequality

$$I_h^0 - \pi I_h^{\max} \leq \theta_h^{\text{out}}.$$

where  $\pi$  is a constant with value ranging from 0 to 1. A higher value of  $\pi$  means less HUB stations and vice versa.

**Unbalanced Stations with Bikes Needed (USBN)** are stations which cannot satisfy user demand at  $\beta_h^-$  service level with their current bike inventory, although they might not be fully

empty. Such station must satisfy the following two inequalities:

$$I_h < I_h^{\min},$$

$$\pi I_h^{\min} - I_h^0 \leq \theta_h^{\text{in}}.$$

The first inequality defines an Unbalanced Station with Bikes Needed while the second makes sure that the station is not an HSBN.

**Unbalanced Stations with Docks Needed (USDN)** are stations which cannot satisfy user demand at  $\beta_h^+$  service level with their current inventory, although they might not be filled to capacity. Such stations must satisfy the following two inequalities:

$$I_h > I_h^{\max},$$

$$I_h^0 - \pi I_h^{\max} \leq \theta_h^{\text{out}}.$$

The first inequality defines an Unbalanced Station with Docks Needed while the second makes sure that the station is not an HSDN.

**Balanced Stations with Bikes Needed (BSBN)** are stations which satisfy the inequality  $I_h \geq I_h^{\min}$ , and hence, are balanced by definition but their current inventory is relatively closer to  $I_h^{\min}$  than  $I_h^{\max}$ . Such stations must satisfy the following two inequalities:

$$I_h \geq I_h^{\min},$$

$$\pi I_h^{\min} - I_h^0 \leq \theta_h^{\text{in}}.$$

**Balanced Stations with Docks Needed (BSDN)** are stations which satisfy the inequality  $I_h \leq I_h^{\max}$ , and hence, are balanced by definition but their current inventory is relatively closer to  $I_h^{\max}$  than  $I_h^{\min}$ . Such stations must satisfy the following two inequalities:

$$I_h \leq I_h^{\max},$$

$$I_h^0 - \pi I_h^{\max} \leq \theta_h^{\text{out}}.$$

Denote each subset of stations by  $\mathcal{S}_{\text{HSDN}}$ ,  $\mathcal{S}_{\text{HSBN}}$ ,  $\mathcal{S}_{\text{USBN}}$ ,  $\mathcal{S}_{\text{USDN}}$ ,  $\mathcal{S}_{\text{BSBN}}$ , and  $\mathcal{S}_{\text{BSDN}}$  respectively. Within each subset, we order stations from the most balanced to the least balanced. We denote the number of stations in each subset by  $|\mathcal{S}_{\text{HSDN}}|$ ,  $|\mathcal{S}_{\text{HSBN}}|$ ,  $|\mathcal{S}_{\text{USBN}}|$ ,  $|\mathcal{S}_{\text{USDN}}|$ ,  $|\mathcal{S}_{\text{BSBN}}|$ , and  $|\mathcal{S}_{\text{BSDN}}|$  respectively.

### 4.2.3 Step 2: Update the Incentives Vector

After the network has been divided into six exclusive sets of stations, its time to update the price change vector based on the following equation:

$$q_{mn}^k = q_{mn}^{k-1} + \Delta q_{mn},$$

where  $\Delta q_{mn}$  is the price update that we will calculate as follows.

For each link  $(m, n)$ , the price change is updated in discrete jumps. We define  $\Delta i_s$  and  $\Delta i_l$  as small and large discrete jumps. Their values are constant numbers. We use \$0.05 and \$0.3 for experimental purposes. As shown in Table 2 the discrete jumps can thus vary from  $-\$0.05$  to  $\$0.05$  when  $\delta_s$  is used or from  $-\$0.3$  to  $\$0.3$  when  $\delta_l$  is used. The trigonometric distribution functions used to calculate  $\Delta q_{mn}$  are plotted in Figure 5. We have not included stations of type BSBN and BSDN in the table because the price change for travel between and to the *balanced* stations is zero. The value of  $\Delta q_{mn}$  between stations of HSBN and USBN types follows the distribution in figure 5a with stations at both ends of the vector getting smaller decrements while those in the middle getting the maximum decrements. This makes sure that our pricing favors the movement of bicycles from stations in the subset  $\mathcal{S}_{\text{HSDN}}$  to the middle stations of the subset  $\mathcal{S}_{\text{USBN}}$ . The reason for favoring middle stations is that the stations at the start are very close to being balanced and hence do not need large changes in price while the stations at the end are very unbalanced and it is difficult to make them balanced again using price changes. Similarly, the value of  $\Delta q_{mn}$  between stations of USBN and USDN types follows the distribution in Figure 5d. The price increment is

Table 2: Calculating values of  $\Delta q_{mn}$ , where  $j_m$  and  $j_n$  are the order indices starting from zero within the set  $\mathcal{S}_{\text{USBN}}$  of stations  $m$  and  $n$ , respectively—for example, if station  $m$  is the most balanced station in the set  $\mathcal{S}_{\text{USBN}}$ , then  $j_m=0$ —and similarly,  $k_m$  and  $k_n$  are the order indices starting from zero within the set  $\mathcal{S}_{\text{USDN}}$ . We let  $\Delta q_{mn} = 0$  for all  $m \in \mathcal{S}_{\text{BSBN}}$  and  $n \in \mathcal{S}_{\text{BSDN}}$ .

Type of Station $m$	Type of Station $n$			
	HSBN	HSDN	USBN	USDN
HSBN	0	$-\Delta i_s$	$-\Delta i_l \sin \frac{\pi j_n}{ \mathcal{S}_{\text{USBN}} }$	$\Delta i_s \sin \frac{\pi k_n}{ \mathcal{S}_{\text{USDN}} }$
HSDN	$\Delta i_s$	0	$-\Delta i_s \sin \frac{\pi j_n}{ \mathcal{S}_{\text{USBN}} }$	$\Delta i_l \sin \frac{\pi k_n}{ \mathcal{S}_{\text{USDN}} }$
USBN	$\Delta i_l \sin \frac{\pi j_m}{ \mathcal{S}_{\text{USBN}} }$	$\Delta i_s \sin \frac{\pi j_m}{ \mathcal{S}_{\text{USBN}} }$	$\frac{\Delta i_s}{2} \left( -\cos \frac{\pi j_n}{ \mathcal{S}_{\text{USBN}} } + \cos \frac{\pi j_m}{ \mathcal{S}_{\text{USBN}} } \right)$	$\Delta i_l \sin \frac{\pi(j_n+k_m)}{ \mathcal{S}_{\text{USBN}} + \mathcal{S}_{\text{USDN}} }$
USDN	$-\Delta i_s \sin \frac{\pi k_m}{ \mathcal{S}_{\text{USDN}} }$	$-\Delta i_l \sin \frac{\pi k_m}{ \mathcal{S}_{\text{USDN}} }$	$-\Delta i_l \sin \frac{\pi(j_n+k_m)}{ \mathcal{S}_{\text{USBN}} + \mathcal{S}_{\text{USDN}} }$	$\frac{\Delta i_s}{2} \left( -\cos \frac{\pi k_n}{ \mathcal{S}_{\text{USDN}} } + \cos \frac{\pi k_m}{ \mathcal{S}_{\text{USDN}} } \right)$

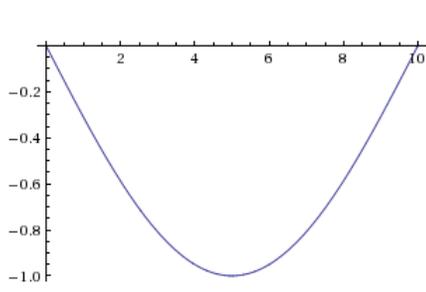
maximum for stations in the middle making it difficult for bicycles to move from stations in subset  $\mathcal{S}_{\text{USBN}}$  to stations in subset  $\mathcal{S}_{\text{USDN}}$ , so on and so forth.

#### 4.2.4 Step 3: Calculate the Number of Unbalanced Stations

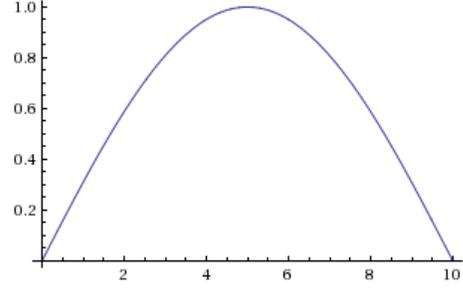
We use (P) to calculate the number of unbalanced stations given the price change vector  $q_{mn}$ . We store this value for every iteration. (P) is also used to calculate the updated values of  $I_h$ . After this, Step 1 is repeated with updated  $I_h$  and stations are again categorized into different types based on Step 1. This process is repeated until the designated number of iterations.

#### 4.2.5 Step 4: Find the Best Price Change Vector

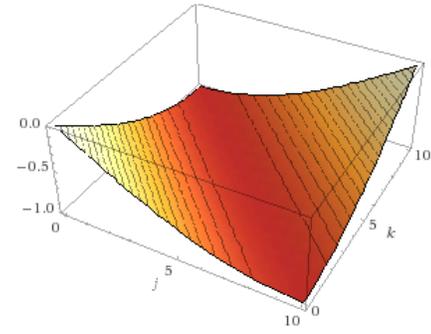
The number of unbalanced stations calculated in Step 3 are compared to the minimum value. If the current price change vector improves on the objective function, the vector and the objective value are stored. After this, Step 0 is repeated with updated  $I_h$  and the values of some of the parameters are calculated based on new  $I_h$ . Stations are again categorized into different types based on Step 1.



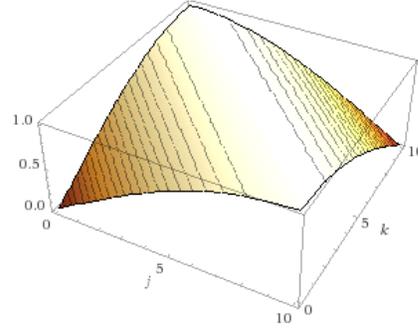
(a)  $-\sin \frac{\pi j}{|\mathcal{S}_{USBN}|}$  for  $j \in [0, |\mathcal{S}_{USBN}| - 1]$



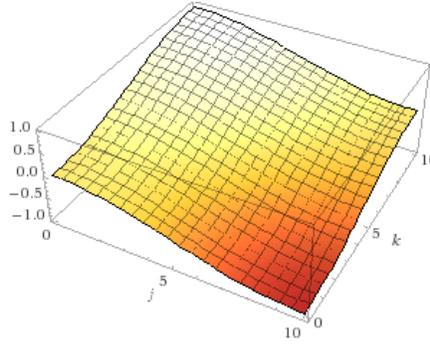
(b)  $\sin \frac{\pi j}{|\mathcal{S}_{USBN}|}$  for  $j \in [0, |\mathcal{S}_{USBN}| - 1]$



(c)  $-\sin \frac{\pi(j+k)}{|\mathcal{S}_{USBN}|+|\mathcal{S}_{USDN}|}$  for  $j \in [0, |\mathcal{S}_{USBN}| - 1]$  and  $k \in [0, |\mathcal{S}_{USDN}| - 1]$



(d)  $\sin \frac{\pi(j+k)}{|\mathcal{S}_{USBN}|+|\mathcal{S}_{USDN}|}$  for  $j \in [0, |\mathcal{S}_{USBN}| - 1]$  and  $k \in [0, |\mathcal{S}_{USDN}| - 1]$



(e)  $(-\cos \frac{\pi j}{|\mathcal{S}_{USBN}|} + \cos \frac{\pi k}{|\mathcal{S}_{USBN}|-1})$  for  $j, k \in [0, |\mathcal{S}_{USBN}| - 1]$

Figure 5: Graphs of Different  $\Delta q_{mn}$  functions for  $|\mathcal{S}_{USBN}| = 10$  and  $|\mathcal{S}_{USDN}| = 10$

Table 3: Journey Data Sample

Start Time	Station ID	End Time	Station ID	Bicycle ID
7/1/2013 12:02:00 AM	31250	7/1/2013 12:21:00 AM	31506	W20231
9/4/2013 4:27:00 PM	31247	9/4/2013 5:11:00 PM	31248	W01484
8/9/2013 3:54:00 PM	31253	8/9/2013 4:01:00 PM	31229	W20198
9/23/2013 5:34:00 PM	31004	9/23/2013 5:52:00 PM	31007	W20523

Table 4: Station Status Data Sample

Station ID	Longitude	Latitude	No of Bicycles	No of Empty Docks
31000	-77.0512	38.8561	4	7
31605	-77.0023	38.8851	1	10
31609	-77.0213	38.8767	7	12
31403	-77.0198	38.9566	8	7

This process is repeated until the designated number of iterations. After the iterations run out, this scheme gives the price change vector with minimum number of unbalanced stations as its output.

## 5 Numerical Experiments

The following publicly available data sets were retrieved from the Capital Bike Share website:

1. Journey Data: The data of 0.85 million bike rides during the three months period starting from July 1st to September 30th, 2013 (See Table 3).
2. Station Data: The data of longitude and latitude of all stations, their ID numbers and their current inventory status retrieved from publicly available xml file (See Table 4).

Journey data is used to calculate the demand vector  $d^{rs}$ . Capital Bike Share is a relatively newer and smaller system. The number of journeys between each OD pair in a small enough time period is very small. So we divide each day into four equal intervals of six hours. We use the data for the first interval of 6:00 am to 11:59 am on 20th September 2013. We assume that the demand in this interval is known at the beginning of the interval. For calculation, we use the demand vector  $d^{rs}$  during this interval using the historical ride data available. The journey data is also used to

determine service level requirements and values of  $I_h^{\max}$  and  $I_h^{\min}$ . We use  $\beta_h^-$  and  $\beta_h^+$  values of 0.85 for our calculations.

Station data with sample in Table 4 is available in the form of an XML file and the current status of the stations in terms of inventory, checks ins and check outs is continuously updated on to the server. This data is used to calculate the position of each station in the network. The longitude and latitude are used to calculate distances and time of travel between all OD pairs. The distance calculated is not euclidean and represents actual biking distance. We also get the number of bicycles, number of empty docks and capacity of each station using this data. The starting inventory  $I_h^0$  used in our calculations is the starting inventory of the network at 9 am on 20th September, 2013. The distance matrix is also used to determine the neighborhood stations of each station. We use 600 meters as a conservative measure with stations less than 600 meters apart becoming each other's neighbors.

We build the data sets for three sets of problems of different network size. A smaller network with 21 closely located stations, a medium sized network with 60 stations and a full sized network with 202 stations in it. It should be taken into account that the number of unbalanced stations cannot always be 0. In fact, our observations show that during high demand periods, like the one we have used for experimentation, the total inventory available in the system is much smaller than demand for the next time period. In Table 5 we present the inventory situation for each of our data sets. As evident, there is always a measure of imbalance present in the system.

The Objective Function values for optimization model and two heuristics for different network sizes are tabulated in Table 6. All the numerical experiments were done on a machine with 2.30 GHz CPU clock speed, 8 GB RAM and 64-bit Windows 8 operating system. The model (P) was solved using the Java API of CPLEX V12.4 while the coding for both GA and IPAS heuristics was also done in Java. We find out that for small sized and medium sized networks, the optimization model is solved by CPLEX within 11 seconds and 1 minute respectively. However, CPLEX fails to solve the full network model consisting of 202 stations. Although the heuristics developed in this paper perform worse of in terms of objective function for the full network, they still show a marked improvement. The columns compare the values of objective function without and with changes in

Table 5: Inventory Status for Different Network Sizes. The three small networks, SMALL-1, SMALL-2, and SMALL-3 have the same network topology and bike demand, but different initial distributions of bike inventory.

Network Name	No of Stations	No of Total Docks	No of Available Bikes	No of Empty Docks	System-wide Demand for Bicycles	System-wide Demand for Docks
SMALL-1	21	443	106	337	107	107
SMALL-2	21	443	190	253	107	107
SMALL-3	21	443	332	111	107	107
MEDIUM	60	1606	1240	366	907	907
LARGE	202	4498	1012	3486	2531	2531

the base price. We see a huge improvement in the objective function when price change vectors are generated through optimization model and heuristics. We also consider the effect of neighborhood distance on the amount of improvement we can bring to our objective function. Since Capital Bike Share is not as big a system as are Vélib in Paris or Citi Bike in New York City, we assume that the number of neighborhood stations with the 600 meter radius is much smaller in Washington DC than in Paris or NYC. We increase the neighborhood radius to 800 meters to better simulate a big city network. With the increase in neighborhood radius, the range of choices available to customers increases and the objective function decreases correspondingly.

We also limit the maximum absolute value of price changes available to the operator. The Table 7 shows the objective function values for different ranges of incentive and penalty values and different  $\nu_3$  values. Smaller values of  $\nu_3$  underestimate the cost of walking and hence alternative paths become more suitable for travelers. We have used  $\nu_3=1/50$  in our earlier calculations.

As evident, larger the price changes an operator is willing to introduce, the more balanced a network becomes. We see that as the price change range increases the total price change offered increases dramatically while offering much less in terms of objective function improvement. It is always better to offer smaller price changes first and gradually increase them if doing so provides considerable advantage.

Table 6: Value of Objective Function for different starting inventories with a) Optimization Model b) Genetic Algorithm and c) Iterative Price Adjustment Scheme (IPAS) for the SMALL-1, SMALL-2, SMALL-3, MEDIUM and LARGE networks

Network Name	Computational Method	Initial Number of Unbalanced Stations	Final Number of Unbalanced Stations	CPU Time (seconds)
SMALL-1	P	9	0	1.609
	GA	9	4	41.275
	IPAS	9	5	3.743
SMALL-2	P	4	0	0.590
	GA	4	0	22.933
	IPAS	4	0	4.034
SMALL-3	P	12	2	0.659
	GA	12	5	39.897
	IPAS	12	6	4.204
MEDIUM	P	25	15	63.314
	GA	25	19	183.325
	IPAS	25	19	9.750
LARGE	P	117	n/a	n/a
	GA	117	94	2288.959
	IPAS	117	84	86.786

Table 7: Value of Objective Function  $f(q_{mn})$  for different  $q_{mn}$  ranges and  $\nu_3$  values where neighborhood radius is 800m

	Range of $q_{mn}$					
	$[-5, 5]$	$[-4, 4]$	$[-3, 3]$	$[-2.5, 2.5]$	$[-2, 2]$	$[-1.5, 1.5]$
$\nu_3=1/25$	79	87	95	103	105	112
$\nu_3=1/50$	70	71	75	76	81	90
$\nu_3=1/75$	64	68	68	71	73	78

Table 8: Comparison of Cost(in Dollars) to the System for Price Induced and Manual Repositioning, A=Total incentives given, B= Total Penalties Levied, C= Cost of Manual Repositioning

Range	$\nu_3=1/25$			$\nu_3=1/50$			$\nu_3=1/75$		
	A	B	C	A	B	C	A	B	C
$q_{mn} \in [-5, 5]$	1311	-1955	744	913	-1426	1183	690	-1102	1148
$q_{mn} \in [-4, 4]$	1251	-1742	605	894	-1361	1134	742	-1095	1235
$q_{mn} \in [-3, 3]$	1039	-1337	399	726	-1078	952	604	-919	1123
$q_{mn} \in [-2.5, 2.5]$	983	-1162	224	668	-973	896	537	-804	1039
$q_{mn} \in [-2, 2]$	820	-954	168	520	-829	822	480	-697	945
$q_{mn} \in [-1.5, 1.5]$	590	-724	63	476	-689	592	376	-577	801

In Table 8, the values in column A represent the total *positive* price change which is the amount of incentives doled out to customers. The values in column B are the total *negative* price change which is the total penalties levied on the traveler to stop them from making undesirable journey choices. Naturally, due to the price change vector, the travelers automatically move certain bikes between stations to make them balanced. The sum of A and B values is the total cost to the system for moving a certain number of bikes automatically. Column C indicates the cost of moving the same number of bikes using manual trucks and crew. Simply, column C is the cost that is saved as a result of pricing scheme. To find the cost values in column C, we use the average \$3.5 as a conservative measure for cost of a single manual repositioning and multiply it with the total number of bicycles moved as a direct result of our pricing scheme. As long as  $A+B \leq C$ , price induced repositioning is more viable than manual repositioning.

Note that if the range of incentives increases, the number of unbalanced stations comes down. Also, the number of bikes ‘automatically’ moved as a direct result of the pricing scheme also increases and hence we see greater values in column A, B and C. Naturally, if the range of price change is larger, greater cost is saved.

## 6 Conclusion

At the start of this paper, we set out to prove the efficacy of a pricing scheme for partially or fully rebalancing a BSS. This paper explores pricing and incentive schemes as a way to rebalance the

network of a public bike sharing system. As already stated, the objective is to minimize the number of unbalanced stations to fully or partially obviate the need for a manual repositioning operation using trucks and crew. We develop a bilevel optimization model that works well for relatively smaller networks to deliver the optimal pricing scheme. We develop heuristics to solve the issue of time while still improving the objective function. A genetic algorithm formulation improves markedly on time but due to very large number of starting chromosomes, does not perform as well with the objective function. We then set out to develop a more intuitive heuristic based on the classification of different stations using the data available. This heuristic called iterative price adjustment scheme delivers much better computing time. For the full network, the IPAS takes only 87 seconds. The value of objective function is also markedly better than genetic algorithm. We conclude that this time is small enough to make it feasible for BSS operators to update their price vector in real time. The cost of offering incentives is also much smaller than the cost reduction from smaller number of unbalanced stations, smaller crew and trucking fleet.

The demand vector we use is based on actual rides data after the fact. One obvious improvement is to model the demand more accurately using demand forecasting taking into account various factors that affect the demand for bicycles. The values of coefficients for deriving cost of travel have also been roughly determined. We assume the value of time to be the same for all customers which is obviously not the case. The values of these coefficients can be better estimated using insights into customer behavior. We conclude that a real time dynamic pricing scheme cannot only solve the problem of system wide imbalance in BSS but also cut down on operating costs of controlling that imbalance.

## **Acknowledgement**

The fourth author's research was supported by the University Transportation Research Center (UTRC) under Grant Number 49997-34-24.

## References

- Benchimol, M., P. Benchimol, B. Chappert, A. De La Taille, F. Laroche, F. Meunier, L. Robinet, et al. 2011. Balancing the stations of a self-service bike hire system. *RAIRO-Operations Research* **45**(1) 37–61.
- Caggiani, L., M. Ottomanelli. 2012. A modular soft computing based method for vehicles repositioning in bike-sharing systems. *Procedia-Social and Behavioral Sciences* **54** 675–684.
- Contardo, C., C. Morency, L.-M. Rousseau. 2012. Balancing a dynamic public bike-sharing system. URL <https://www.cirrelt.ca/DocumentsTravail/CIRRELT-2012-09.pdf>.
- DeMaio, P. 2009. Bike-sharing: History, impacts, models of provision, and future. *Journal of Public Transportation* **12**(4) 41–56.
- Fricker, C., N. Gast. 2014. Incentives and redistribution in homogeneous bike-sharing systems with stations of finite capacity. *EURO Journal on Transportation and Logistics* 1–31.
- Haupt, R. L., S. E. Haupt. 2004. *Practical genetic algorithms*. John Wiley & Sons.
- Meddin, R., P. DeMaio. 2014. URL <http://www.bikesharingworld.com>. Accessed on September 19, 2014.
- New York City Department of City Planning. 2009. Bike-share opportunities in New York City. [http://www.nyc.gov/html/dcp/pdf/transportation/bike\\_share\\_complete.pdf](http://www.nyc.gov/html/dcp/pdf/transportation/bike_share_complete.pdf).
- Pfrommer, J., J. Warrington, G. Schildbach, M. Morari. 2014. Dynamic vehicle redistribution and online price incentives in shared mobility systems. *Intelligent Transportation Systems, IEEE Transactions on* **15**(4) 1567–1578.
- Raviv, T., M. Tzur, I. A. Forma. 2013. Static repositioning in a bike-sharing system: models and solution approaches. *EURO Journal on Transportation and Logistics* **2**(3) 187–229.
- Schuijbroek, J., R. Hampshire, W.-J. van Hoesve. 2013. Inventory rebalancing and vehicle routing in bike sharing systems. URL <http://repository.cmu.edu/tepper/1491/>.

- Shaheen, S. A., S. Guzman, H. Zhang. 2010. Bikesharing in europe, the americas, and asia. *Transportation Research Record: Journal of the Transportation Research Board* **2143**(1) 159–167.
- Shu, J., M. C. Chou, Q. Liu, C.-P. Teo, I.-L. Wang. 2013. Models for effective deployment and redistribution of bicycles within public bicycle-sharing systems. *Operations Research* **61**(6) 1346–1359.
- The Pennsylvania Environmental Council. 2013. Philadelphia bike share strategic business plan. URL <http://www.bikesharephiladelphia.org/PhilaStudy/CompleteBusinessPlan.pdf>.
- Waserhole, A., V. Jost, et al. 2012. Vehicle sharing system pricing regulation: A fluid approximation. URL <http://hal.archives-ouvertes.fr/hal-00727041/>.

# Appendix

## A Genetic Algorithm

A brief description of the algorithm steps is presented first. Let us define a chromosome as an array of variable values to be optimized. In our case,  $q_{mn}$ , the price change vector is the chromosome. Each chromosome of the population is a  $1 \times N_{\text{var}}$  vector. In our case with  $n$  stations, there are  $n^2$  price change variables and hence each chromosome is an array of size  $1 \times n^2$ . Let us also convert variables  $q_{mn}$  to a single subscript vector for convenience. Let  $\chi_l$  represent the individual variables of the chromosome vector  $q_{mn}$  where  $l$  varies from 1 to  $N_{\text{var}}$ ,

$$\text{chromosome} = [q_{m_1n_1}, q_{m_1n_2}, \dots, q_{m_1n_n}, \dots, q_{m_nn_n}] = [\chi_1, \chi_2, \chi_3, \dots, \chi_{N_{\text{var}}}] \quad (31)$$

Each chromosome thus constructed has a cost associated with it through a cost function. In our case the upper level objective, i.e., the number of unbalanced stations, is the cost function. This number depends on the values of variables  $x_{mn}^{rs}$ , which in turn depend on the values  $c_{mn}^{rs}$ , and hence,

$q_{mn}$  or  $\chi_l$ ,

$$\text{Number of unbalanced stations} = f(\text{chromosome}) = f([\chi_1, \chi_2, \chi_3, \dots, \chi_{N_{\text{var}}}] \quad (32)$$

### A.1 Initial Population

The initial population consists of a number of chromosomes equal to the population size  $N_{\text{size}}$ . The population size is determined keeping in mind the size of each chromosome and the execution time to find the cost function. We choose an initial population of 300 chromosomes. The initial population matrix thus consists of  $N_{\text{size}}$  number of chromosomes, each representing a row of  $N_{\text{var}}$  continuous values. These values are randomly generated and lie within a pre-specified range,

$$(\chi_l^{\text{lo}}) \leq \chi_l \leq (\chi_l^{\text{hi}}).$$

We use -3 and 3 for  $\chi_l^{\text{lo}}$  and  $\chi_l^{\text{hi}}$ , respectively. Let  $\zeta$  be a random real number from a uniform distribution between 0 and 1. The value of each variable  $\chi_l$  in a single chromosome is generated by:

$$\chi_l = \chi_l^{\text{lo}} + \zeta(\chi_l^{\text{hi}} - \chi_l^{\text{lo}}).$$

The initial chromosome population thus consists of  $N_{\text{size}}$  chromosomes, where in turn, each chromosome consists of  $N_{\text{var}}$  randomly generated values of variables  $\chi_l$ .

### A.2 Natural Selection

The cost function for each chromosome in the initial population is evaluated for the number of unbalanced stations. At this point, all the chromosomes are ranked in increasing order of their cost function values obtained via equation (32). The chromosome with the minimum number of unbalanced stations comes at the top and so forth. After ranking, the top-ranked 50% of the chromosomes are kept while the rest of them are discarded to make room for new offspring.

### A.3 Pairing and Mating

After ranking the population of chromosomes and selecting the top 50% contenders, the chromosomes are broken into pairs to mate and generate new offspring. To make sure that the characteristics of the best chromosomes, i.e., those with the least number of unbalanced stations are transferred to the offspring, we use Rank Weighting method to set the probability of mating. This method assigns higher probabilities to the mating of the chromosomes with lower values of the cost function.

Let  $a$  be the rank index of the top 50% chromosomes after they have been ranked in increasing order of the cost function. Lower the rank  $a$  of a chromosome, lower the value of its cost function. For a chromosome with rank  $a$ , the mating probability  $P_a$  is set as

$$P_a = \frac{N_{\text{Keep}} - a - 1}{\sum_{a=1}^{N_{\text{Keep}}} a},$$

where  $N_{\text{Keep}}$  is the number of chromosomes selected to pair and mate which is equal to  $\lfloor N_{\text{size}}/2 \rfloor$ . The mating probabilities thus found are then used to calculate cumulative probability  $P_a^{\text{cum}}$  for each chromosome using the equation,

$$P_a^{\text{cum}} = P_{a-1}^{\text{cum}} + P_a \quad \forall a \in [1, N_{\text{Keep}}],$$

where  $P_0^{\text{cum}} = 0$ . To produce  $\lfloor N_{\text{size}}/2 \rfloor$  new chromosomes,  $\lfloor N_{\text{size}}/2 \rfloor$  number of parents is selected out of which  $\lfloor N_{\text{size}}/4 \rfloor$  are *father* and  $\lfloor N_{\text{size}}/4 \rfloor$  are *mother* chromosomes. The selection of parent chromosomes to mate and reproduce is done using the following procedure:

```

while  $c = 1$  to  $\lfloor N_{\text{size}}/4 \rfloor$  do
    ▷ Generate two random real numbers in the range  $[0, 1]$  and call them  $\text{Pick}_c^1$  and  $\text{Pick}_c^2$ 
for  $a = 1$  to  $\lfloor N_{\text{size}}/2 \rfloor$  do
    if  $\text{Pick}_c^1 \leq P_a^{\text{cum}}$  and  $\text{Pick}_c^2 \geq P_{a-1}^{\text{cum}}$  then
        ▷ Select the chromosome with rank  $a$  to be the mother and call it  $\text{ma}_c$ 
    if  $\text{Pick}_c^2 \leq P_a^{\text{cum}}$  and  $\text{Pick}_c^1 > P_{a-1}^{\text{cum}}$  then

```

▷ Select the chromosome with rank  $a$  to be the *father* and call it  $pa_c$

**end if**

**end if**

**end for**

▷ Use the pair  $ma_c$  and  $pa_c$  to mate and produce a pair of offspring chromosomes

**end while**

For mating, the algorithm we use is taken from Haupt and Haupt (2004) and is a combination of an extrapolation method with a crossover method as described henceforth. We generate a random number  $\eta$  in range  $[0, N_{\text{var}}]$  to select a variable in the first pair of parent chromosomes to be a crossover point,

$$\eta = \lceil \zeta N_{\text{var}} \rceil,$$

where  $\zeta$  is a random number from a uniform distribution between 0 and 1 and  $\lceil \cdot \rceil$  is a ceiling function. The crossover point  $\eta$  can be any number between 1 and  $N_{\text{var}}$ . The variables before the crossover point retain their original value while those after the crossover point are combined to form new variables that will appear in the children. Following equations are used to generate new variables. Let us represent the two parents as

$$\begin{aligned} \text{mother} &= [\chi_{m1}, \chi_{m2}, \dots, \chi_{m\eta}, \dots, \chi_{mN_{\text{var}}}] \\ \text{father} &= [\chi_{d1}, \chi_{d2}, \dots, \chi_{d\eta}, \dots, \chi_{dN_{\text{var}}}] \end{aligned}$$

where the subscripts  $m$  and  $d$  represent mother and father. These two parent chromosomes will then combine to produce variables for the two offspring chromosomes using the extrapolation method,

$$\begin{aligned} \chi_{\text{new1}} &= \chi_{m\eta} - \phi[\chi_{m\eta} - \chi_{d\eta}], \\ \chi_{\text{new2}} &= \chi_{d\eta} + \phi[\chi_{m\eta} - \chi_{d\eta}]. \end{aligned}$$

Where  $\phi$  is a random number from a uniform distribution between 0 and 1. Finally, two offspring

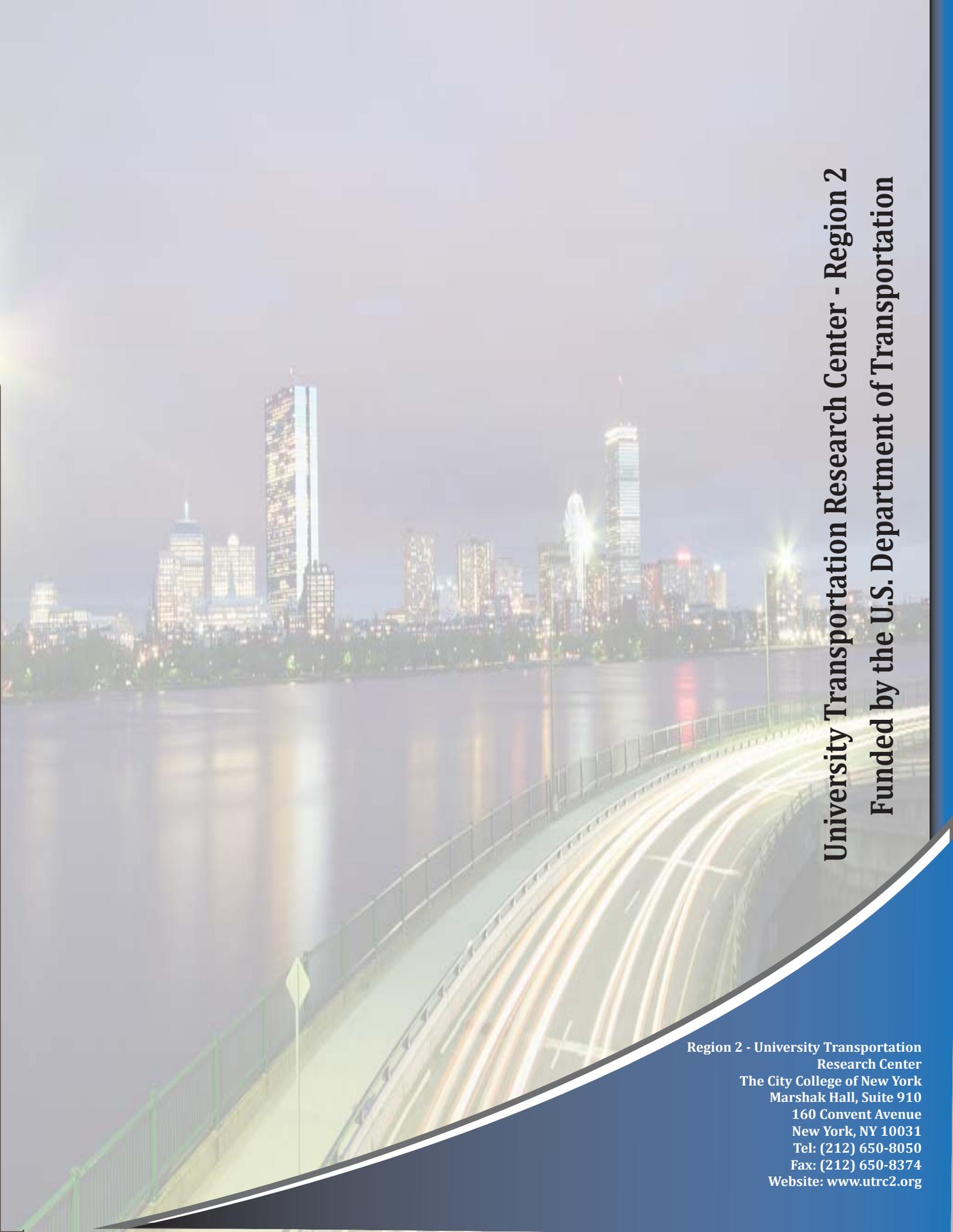
chromosomes are created through crossover of variables after  $\chi_\eta$  in the parent chromosomes using a crossover method,

$$\text{offspring}_1 = [\chi_{m1}, \chi_{m2}, \dots, \chi_{m,\text{new}}, \dots, \chi_{dN_{\text{var}}}],$$

$$\text{offspring}_2 = [\chi_{d1}, \chi_{d2}, \dots, \chi_{d,\text{new}}, \dots, \chi_{mN_{\text{var}}}]$$

#### A.4 Next Generation and Convergence of the GA Heuristic

After producing the offspring chromosomes, shape of the original population is modified. The population now consists of the top 50% parent chromosomes and 50% offspring chromosomes. When this new population is generated, we go back to natural selection and repeat all the subsequent steps. This process continues over a number of iterations. Better and better chromosomes with fewer number of unbalanced stations find their way to the top of ranking. This process is continued until the cost value of the best identified solution converges to a certain number and the successive iterations stop producing any improvement. In our case, instead of defining a convergence criterion, we let the algorithm run for a certain number of iterations and save the value of the top-ranked chromosome. After the algorithm stops, the result is a price change vector chromosome that gives the minimum number of unbalanced stations.

A long-exposure photograph of a city skyline at night, reflected in a body of water. In the foreground, a bridge or highway has light trails from moving vehicles. The sky is dark, and the city lights are bright and colorful.

**University Transportation Research Center - Region 2**  
**Funded by the U.S. Department of Transportation**

**Region 2 - University Transportation  
Research Center**  
**The City College of New York**  
**Marshak Hall, Suite 910**  
**160 Convent Avenue**  
**New York, NY 10031**  
**Tel: (212) 650-8050**  
**Fax: (212) 650-8374**  
**Website: [www.utrc2.org](http://www.utrc2.org)**