Proceedings of the 2017 Winter Simulation Conference
W. K. V. Chan, A. D'Ambrogio, G. Zacharewicz, N. Mustafee, G. Wainer, and E. Page, eds.

DATA-DRIVEN ADAPTIVE THRESHOLD CONTROL FOR BIKE SHARE SYSTEMS

Felisa J. Vázquez-Abad
Silvano Bernabel

Department of Computer Science, Hunter College,
and CUNY Institute for Computer Simulation,
Stochastic Modeling and Optimization,
New York, NY 10065, USA

Michael C. Fu

Robert H. Smith School of Business,
Institute for Systems Research
University of Maryland
College Park, MD 20742, USA

ABSTRACT

When it comes to building a successful public bike share such as the NYC CitiBike system, there are many important questions to answer. There is no shortage of work being done on finding the most efficient way to redistribute bikes to stations. Equally important to how to distribute bikes is the question of when to redistribute bikes. Redistribute too infrequently and customers become frustrated, resulting in decreased revenue. Redistribute too frequently and the cost of redistribution becomes prohibitively high. In this piece of research, we attempt to find the optimal time to call for the redistribution of bikes to minimize cost and retain maximum membership.

1 INTRODUCTION

The past decade has seen a rapid increase in the deployment of massively used bike share systems (BSS) stations all over the World (NYCDP 2009). It is believed that shared vehicle transportation is a much needed sustainable alternative for city dwellers in the 21st Century (Bullock, Brereton, and Bailey 2017). Today public transportation is perhaps one of the most exciting areas in society where massive amounts of data is being collected. But there is yet no consensus as to how this data should be used. Our focus is on one-way vehicle share systems and we will heavily draw on the bike share example, but our algorithms can be adapted to other vehicles, and ultimately link with multi-modal transportation. This paper focuses on the use of streaming data in order to apply the appropriate adaptive control for balancing the usage of the vehicles.

The so-called "one way vehicle sharing systems" pose the problem that the vehicles can be taken from their origin and returned to a different station as destination. This naturally creates temporary unbalancing of resources, so that vehicles are missing in popular origin stations and docks are full in popular destinations (Bruglieri, Colorni, and Luè 2014). The redistribution problem is a scheduling problem that determines how many vehicles should be taken from each station and brought to another station. It is a complex problem and one that admits many optimal solutions, so literature abounds on doing the redistribution taking account of matters such as the truck capacity for carrying the vehicles, minimizing distance traveled by each re-distributed vehicle, etc. (Chemla et al. 2013, Chemla et al. 2013, Raviv et al. 2013, Schuijbroek et al. 2017, Shu et al. 2013). The problem is also referred to as "rebalancing" problem.

There are two different problems: the static problem seeks to find the optimal initial distribution and move vehicles overnight when the system is closed to the public and traffic is light. The dynamic problem tries to solve the scheduling during the day when the system is in demand. For the dynamic allocation problem, graph-theoretic methods have been used to determine which bikes to take from a station and deposit in another one. For example, Singhvi et al. (2015) focus on finding local solutions within neighboring stations via graph-theoretic arguments to pair vertices on a graph, using integer programming and feeding the model with peak-hour historical data. We look at a related but different problem for dynamic redistribution: not the "how", but the "when" should dynamic redistribution be scheduled.

Our model assumes that each station undergoes a number of "regimes" during each day. The rate of arrival of passengers and the rate of arrival of vehicles is constant for each regime, but change dramatically from one regime to the next. This approach is consistent with the classification of peak and off-peak hours that results from regular work and leisure patterns, driving the demand for transportation. Section 2 presents the stationary model for one station. Section 3 describes the optimal call for redistribution for each regime, based on a stochastic model and the corresponding stationary control. Section 4 introduces a more realistic non-stationary model and describes the results of the simulations adapting the control policies of Section 3. In Section 5 we study a different approach that incorporates streaming information from users.

2 STATIONARY MODEL

2.1 Birth and Death Model for Station Occupancy

Assume that for each origin-destination pair (o,d) the arrival process of clients at station o requesting a bike for destination d is a Poisson process with rate $\lambda_{o,d}$. Furthermore we assume that these processes are independent of each other. Consecutive travel times associated with the O-D pair (o,d) are iid (independent and identically distributed) random variables with distribution G of mean $T_{o,d}$.

For any given origin o, people arriving that wish to take bikes may or may not find one available. Call $X_o(t)$ the number of bikes parked at station o and let C_o denote the station's total capacity (number of docks, assumed constant here). In stationary operation, because Poisson arrivals see a typical stationary state (PASTA) then the *effective* arrival rate is $\lambda'_{o,d} = (1-B_o)\,\lambda_{o,d}$ where $B_o = \mathbb{P}[X_0(t)=0]$. It follows that the process of departure of bikes is a Poisson process with rate $\lambda'_{o,d}$. In what follows we will use these facts to model the occupation process at each station.

Lemma 1 Fix the O-D pair (o,d). The stationary number of bikes on route from o to d has a Poisson distribution with rate $\lambda'_{o,d}T_{o,d}$. Furthermore, the arrival of bikes from o at destination d follows a Poisson process with rate $\lambda'_{o,d}$.

Proof. The proof of the Lemma follows closely the method by Taylor and Karlin (1998), pages 301-303. It is first established that the number of bikes on route at time t, called $M(t)$ satisfies:

$$\mathbb{P}(M(t)=m) = e^{-\lambda'_{o,d}t\,p(t)}\frac{(\lambda'_{o,d}t\,p(t))^m}{m!}$$

where $t\,p(t) = \int_0^t (1-G(u))\,du \to T_{o,d}$. This implies that the process $M(t)$ has limiting stationary distribution Poisson$(\lambda'_{o,d}T_{o,d})$. The second assertion follows from the observation that the process of arrivals of bikes at d has the same distribution as the departure process of a $M/G/\infty$ queue, which is a Poisson process with rate $\lambda'_{o,d}$. □

Single Station Stationary Model. For the remainder of the paper, and unless otherwise specified, we fix the station and drop subindices to make notation simpler. From Lemma 1, the model that we consider to describe the number of bikes $X(t)$ at time t of a given station under the stationary model is a Birth and Death process. The birth rate λ is the total arrival rate of bikes and the death rate μ is the arrival rate of customers that wish to take bikes at the station.

Lemma 2 Assume that $\lambda < \mu$ and C is total capacity. Let

$$\tau_C(k) = \min(s\colon X(t+s)=0\,|\,X(t)=k) \tag{1}$$

denote the time until starvation of the station, and $T_C(k) = \mathbb{E}[\tau_C(k)]$, then

$$T_C(k) = A\left(\left(\frac{\mu}{\lambda}\right)^k - 1\right) + \frac{k}{\mu-\lambda},\; A = -\left(\frac{\lambda}{\mu}\right)^C \frac{\lambda}{(\mu-\lambda)^2}. \tag{2}$$

Proof. Embed the B&D process $\{X(t)\}$ into a random walk $\{\xi_n\}$ using uniformization, where $\mathbb{P}(\xi_n = i+1 \,|\, \xi_n = i) = p = 1 - \mathbb{P}(\xi_{n+1} = i-1 \,|\, \xi_n = i)$ for $0 < i < C$, plus the reflecting boundary dynamics

$$\begin{aligned}\mathbb{P}(\xi_{n+1} &= C \,|\, \xi_n = C) = p = 1 - \mathbb{P}(\xi_{n+1} = C-1 \,|\, \xi_n = C),\\ \mathbb{P}(\xi_{n+1} &= 1 \,|\, \xi_n = 0) = p = 1 - \mathbb{P}(\xi_{n+1} = 0 \,|\, \xi_n = 0),\end{aligned}$$

with $p = \lambda/(\lambda+\mu)$.

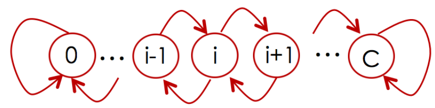

Figure 1: Schematic representation of the embedded random walk $\{\xi_n\}$.

Let $u(k)$ be the expected number of steps until $\xi_n = 0$ given $\xi_0 = k$, then $T_C(k) = u(k)/(\lambda+\mu)$, because the (continuous) time between any two steps is exponentially distributed with rate $\lambda+\mu$.

Calculating $u(k)$ is done now with a modified Gambler's ruin problem. From the dynamics of the random walk it follows that $u(k)$ satisfies the recurrence

$$u(k) = 1 + pu(k+1) + (1-p)u(k-1); \quad 0 < k < C$$

with boundary conditions $u(0) = 0$ and $u(C) = 1 + pu(C) + (1-p)u(C-1)$, where $p = \lambda/(\lambda+\mu)$.

The characteristic polynomial for the homogeneous recurrence is $px^2 - x + (1-p) = 0$ and a particular solution to the inhomogeneous equation is $\tilde{u}(k) = k/(1-2p)$. So the general solution has the form

$$u(k) = a\left(\frac{1-p}{p}\right)^k + b + \frac{k}{1-2p}.$$

Using $u(0) = 0$ yields $a = -b$. The second boundary condition yields the value of the constant a:

$$u(C) = a\left(\frac{1-p}{p}\right)^C - a + \frac{C}{1-2p} = \frac{1}{1-p} + a\left(\frac{1-p}{p}\right)^{C-1} - a + \frac{C-1}{1-2p},$$

which implies

$$a = -\frac{p^{C+1}}{(1-p)^C(1-2p)^2} = -\left(\frac{\lambda}{\mu}\right)^C \frac{\lambda(\lambda+\mu)}{(\mu-\lambda)^2}.$$

Use now $1-2p = (\mu-\lambda)/(\lambda+\mu)$ to get

$$u(k) = a\left(\frac{1-p}{p}\right)^k - a + \frac{k(\lambda+\mu)}{\mu-\lambda},$$

which proves the claim, identifying $a = (\lambda+\mu)A$. □

REMARK. By symmetry, it follows that the expected time to reach the maximal capacity C has a similar expression with the values λ and μ interchanged.

2.2 Optimal Threshold Policies

Customer dissatisfaction arises when a customer wishes to take a bike at an empty station or when a customer fails to find a dock in the desired destination station. We present a solution of the optimization problem that seeks to minimize cost of redistribution given a constraint of the form "at least a fraction $(1-\alpha)$ of clients are satisfied". Special trucks, or sometimes bikes with trailers, bring bikes to the stations that require replenishment, or are ready to take out bikes when there is surplus at the station (or both).

Most of the literature on redistribution addresses the scheduling of these trucks. Instead, here we assume that the scheduling of redistribution trucks is optimal once the call for redistribution is placed, and we try to find the optimal time when calls for redistribution are placed at each station. Fix a station and assume that it follows the stationary model. In our (simple) model, the time that it takes the trucks to arrive at the station for redistribution is a constant ρ. We use this as a first model to simplify the presentation of the control rules, but our algorithms can be modified to include a random time.

The model under control is an inventory-like B&D process, where one specifies the time when the "order" is placed. An order in our model is either an order to replenish for bikes (in case that $\lambda < \mu$) or an order to remove bikes (when $\lambda > \mu$). Because of symmetry of the model (time-reversibility) it suffices to focus on the optimal ordering policies for the "starvation" regimes ($\lambda < \mu$), and apply the mirror policies for the filling regimes. Order deliveries are assumed to re-establish the number of bikes at the station to a fixed value C_0 (either full station, or empty, depending on the regime).

Let $X(t)$ denote a B&D process with constant rates λ, μ and let $\eta(t)$ be the time that the station is empty during a period of time of length t. Specifically,

$$\eta(t) = \int_0^t \mathbf{1}_{\{X(s)=0\}}\, ds.$$

Let κ be the fixed cost for an order delivery (replenishment or removal of bikes). For the stationary model, the goal is to define a Markovian (state dependent) control strategy parametrized by $\theta \geq 0$ that will identify the optimal moments to call for redistribution. To emphasize the dependency on the control, we now use the notation $X_\theta(t)$ for the station occupation process under control θ, and we use $\{\nu_k(\theta), k \geq 1\}$ for the consecutive times between two deliveries. Because the model for $X_\theta(\cdot)$ is a CTMC, the process $\{X_\theta(t); \nu_k(\theta) \leq t < \nu_{k+1}(\theta)\}$ has the same distribution as the process $\{X(t), 0 \leq t < \nu_1(\theta) \,|\, X(0) = C_0\}$. That is, $X_\theta(\cdot)$ is a regenerative process with iid regeneration times $\{\nu_k(\theta), k \geq 1\}$.

We seek the solution to the optimization problem

$$\min_\theta \left(J(\theta) \stackrel{\text{def}}{=} \frac{\kappa}{L(\theta)} \right) \quad \text{s.t. } \frac{I(\theta)}{L(\theta)} \leq \alpha, \tag{3}$$

where $L(\theta) = \mathbb{E}[\nu(\theta) \,|\, X(0) = C_0]$ is the expected time between order deliveries, and $I(\theta) = \mathbb{E}[\eta(\nu(\theta)) \,|\, X(0) = C_0]$ is the expected time during the cycle that the station is empty. The cost function is the stationary average cost of redistribution, and the constraint is the one on the fraction of dissatisfied customers (because customer arrivals follow a homogeneous Poisson process with rate μ). Finally, call $R(\rho) = \mathbb{E}[\eta(\rho) \,|\, X(0) = 0]$ the expected amount of empty time of the B&D process during an interval of length ρ, starting empty.

We propose the following threshold-like policies, adapted to the class of regime:

1. **Super-starvation (SS).** When $\lambda < \mu$ and $R(\rho) > \alpha(T_C(C) + \rho)$. This means that even calling the truck as soon as the station is empty will not satisfy the quality of service constraint. In this case we use a threshold policy of the inventory type: call for redistribution at time $\min(t : X_\theta(t) \leq \theta)$ and the next cycle starts ρ unit of time later. The control here is an integer, $\theta \in \mathbb{N}$.
2. **Starvation (S).** When $\lambda < \mu$ and $R(\rho) \leq \alpha(T_C(C) + \rho)$, but $\lim_{t\to\infty} \mathbb{E}[\eta(t)]/t > \alpha$. This means that calling immediately upon emptying of the station satisfies the constraint, but never calling for an order does not. In this case we use the "wait time" policy: call for redistribution at the k-th cycle at the moment $M = \min(t \geq \nu_{k-1}(\theta) : \eta(t) \geq \theta)$ and $\nu_k(\theta) = M + \rho$. Here the control $\theta \in \mathbb{R}^+$.
3. **Neutral (N).** When $\lambda < \mu$ and $\lim_{t\to\infty} \mathbb{E}[\eta(t)]/t \leq \alpha$. This means that the station is balanced and there is no need for redistribution.

Analogously, if $\lambda > \mu$ we consider the times for filling up to capacity, and $I(t)$ is replaced by the time that the station is full. This will extend the Neutral regime for $\lambda > \mu$, then there is a Filling (F) regime, and finally a Super-Filling (SF) regime. The threshold policies for these regimes will be mirror policies to the ones for $\lambda < \mu$, so it suffices to solve the problems for this latter case.

Lemma 3 Refer to equation (2). Under the control policy for SS regime with given $\theta < C_0$,

$$L(\theta) = T_{C_0-\theta}(C_0-\theta)+\rho, \quad \text{and} \quad I(\theta) = \mathbb{E}[R(\rho-\tau_{C_0}(\theta))\mathbf{1}_{\{\tau_{C_0}(\theta)<\rho\}}]. \tag{4}$$

Under the control policy for the S regime,

$$L(\theta) = T_{C_0}(C_0)+\lambda\theta\, T_{C_0}(1)+\theta+\rho, \quad \text{and} \quad I(\theta) = \theta+\mathbb{E}[R(\rho)]. \tag{5}$$

Proof. Consider first the SS regime's policy. Here we call for redistribution at a time $M = \min(t\colon X(t) \leq \theta \,|\, X(0) = C_0)$. We want to calculate the time it takes from $X(0) = C_0$ until the station reaches level θ for the first time. Shift now the process by $-\theta$: $Y(t) = X(t)-\theta$ (a.s). Then $\{Y(t); 0 \leq t < M\}$ has the same distribution as a B&D process with rates λ,μ but with total capacity also shifted by $-\theta$, which establishes that $M \stackrel{\text{d}}{=} \tau_{C_0-\theta}(C_0-\theta)$. At time M the call is placed and it takes ρ units of time for the redistribution to happen, which establishes (4). Because the truck is called when there are θ bikes, the time to empty is $\tau_{C_0}(\theta)$, and the idle time depends on whether $\tau_{C_0}(\theta)$ is larger or smaller than ρ, yielding the expression for $I(\theta)$.

Consider now the S regime's policy. Starting at $X(0) = C_0$ the cycle evolves as an "on-off" process indicating if there are bikes in the station (on) or if the station is empty (off), as illustrated in Figure 2. Call $\{(\xi_k,\zeta_k)\}$ the consecutive on-off times. By construction $\xi_1 \stackrel{\text{d}}{=} \tau_{C_0}(C_0)$ and for all other $k > 1$, $\xi_k \stackrel{\text{d}}{=} \tau_{C_0}(1)$ (until the truck arrives). Every time that the station empties, the time until the first bike arrival is an exponential random variable with rate λ, so that $\zeta_k \sim \text{Exp}(\lambda)$. Let $N(\cdot)$ be the Poisson process with inter arrival times $\{\zeta_k\}$. Then the number of on-off cycles before time M is

$$\min\left(n\colon \sum_{k=1}^{n}\zeta_k > \theta\right) \stackrel{\text{d}}{=} N(\theta)+1, \qquad N(\theta) \sim \text{Poisson}(\lambda\theta).$$

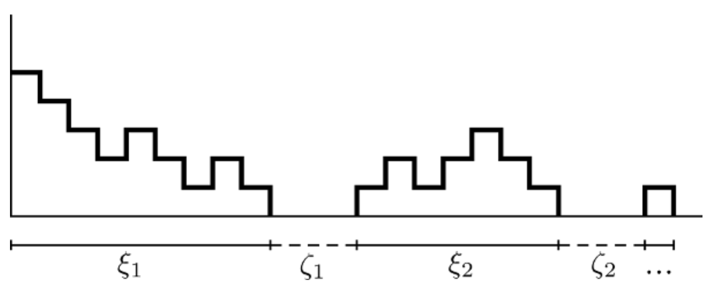

Figure 2: Schematic representation of the occupation process at a station.

It follows that

$$L(\theta) = T_{C_0}(C_0)+\mathbb{E}\left[\sum_{k=2}^{N(\theta)+1}\xi_k\right]+\theta+\rho,$$

and using the fact now that ξ_k are independent of the Poisson process $N(\cdot)$, we obtain (5). By construction, the cumulative idle time until calling for redistribution is θ, and it takes ρ units of time for the truck to arrive, yielding the expression for $I(\theta)$. □

Our final remark in this section is that, because of the on-off structure of the process after an initial time $\tau_C(C)$, $\lim_{t\to\infty}\mathbb{E}[\eta(t)]/t = \lambda^{-1}/(\lambda^{-1}+T_C(1))$. This helps for quick identification of regime N.

3 OPTIMAL STATIONARY POLICIES

3.1 Optimal SS Policy

For the SS regime, the control value is an integer $\theta \in \mathbb{N}$. By inspection, it can be argued that $J(\theta)$ is increasing in θ. The ratio $I(\theta)/L(\theta)$ of dissatisfied customers is a decreasing function of θ, so that the optimal solution of the constrained problem must satisfy:

$$\theta^* = \min(\theta : I(\theta) \leq \alpha L(\theta)).$$

Notice that if $I(C_0)/L(C_0) > \alpha$ then the system is not well designed: ρ is too big to provide the required quality of service and all possible solutions are infeasible. We will assume here that we operate under the case that the dissatisfaction level α can be achieved. We do not have a closed form expression for $I(\theta)$ (unless $\rho = 0$) so it must be estimated via simulations. Solving for θ^* can be done with stochastic binary search (Vázquez-Abad and Fenn 2016).

Algorithm 1.

Initialize $\ell_0 = 0, r_0 = C_0$, $N = N_0$, $n = 1$, a "tolerance" value for the error and a confidence level q.

Step 1. $\theta_n = \left\lfloor \frac{\ell_n + r_n}{2} \right\rfloor$

Step 2. Simulate N cycles to produce iid sample of the empty time $\{\chi_j, j = 1, \ldots N\}$ and evaluate $\hat{I}(\theta_n) = \frac{1}{N}\sum_{j=1}^{N} \chi_j$, $\hat{\sigma}^2 = \frac{1}{N-1}\sum_{j=1}^{N}(\chi_j - \hat{I}(\theta_n))^2$, $\varepsilon = t_{N-1,q}\sqrt{\hat{\sigma}^2/N}$.

if $\hat{I}(\theta_n) - \varepsilon > \alpha L(\theta_n)$ **then** $\ell_n = \theta_n$; $r_n = r_{n-1}$;
else if $\hat{I}(\theta_n) + \varepsilon < \alpha L(\theta_n)$ **then** $\ell_n = \ell_{n-1}$; $r_n = \theta_n$;
else if $\varepsilon <$ tolerance **then return** θ_n;
else
    $N = 2N$; **goto** Step 2;
end if
if $r_n - \ell_n \leq 1$ **then** $\theta_n = r_n$;
    return θ_n;
else goto Step 1;
end if

By construction, the algorithm above finds the closest value (at an approximate (1-q)100% level of confidence, e.g, $t_{N-1,0.01}$ = 2.576 for a 99% level and large N) to the target, accepting an error within the specified tolerance. In (Vázquez-Abad and Fenn 2016) a backtracking modification is introduced that can be useful to correct errors in early discarding. The probability of error is also calculated in that reference.

3.2 Optimal S Policy

For the S regime, $J(\theta)$ is decreasing in $\theta \in \mathbb{R}$, and the ratio $I(\theta)/L(\theta)$ of dissatisfied customers is an increasing function of θ. Given the conditions on the parameters, the solution must satisfy $I(\theta^*) = \alpha L(\theta^*)$ and target tracking can be used to solve the problem. Because the control is continuous (a time-out policy), a binary search is not appropriate, and one may explore other root finding algorithms such as Golden ratio.

We do not have a closed form solution for $I(\theta)$ (unless $\rho = 0$) for the S regime, but we know that $I(\theta) = \theta + R(\rho)$, and $R(\rho)$ is independent of θ, because it corresponds to the expectation of the total time that a station is empty during a period of time of length ρ, starting empty. By elementary statistics, using a large number N of independent simulations, we can estimate $R(\rho)$ by the sample average $\hat{R}(\rho)$, and it follows that $\beta = \hat{R}(\rho) - R(\rho) \approx \text{Normal}(0, S_R^2/N)$, where S_R^2 is the variance of the empty time $\eta(\rho)$ when the process starts at $X = 0$. The algorithm is

$$\theta_{n+1} = \theta_n - \varepsilon_n \left(\frac{\theta_n + \hat{R}(\rho)}{L(\theta_n)} - \alpha \right),$$

where $\{\varepsilon_n\} > 0$ is the step size sequence. The function $L(\theta)$ is given by (5).

Theorem 1 Call c_1, c_2 the constants in (5) such that $L(\theta) = c_1 + c_2\theta$. Assume that $\varepsilon_n \to 0$, $\sum_n \varepsilon_n = +\infty$ and $\sum_n \varepsilon_n^2 < \infty$. Then the algorithm above satisfies $\theta_n \to \bar{\theta}$, where $\bar{\theta} - \theta^* \sim$ Normal$(0, C^2 S^2/N)$, with constant $C = L(\theta^*)/(c_2 R(\rho) - c_1)$.

Proof. It follows from Theorem 2.3 and Lemma 2.1 of Vázquez-Abad and Heidergott (2017) that, given $\hat{R}(\rho)$, the algorithm above is a deterministic descent algorithm that converges to the limit of the ODE

$$\frac{d\theta(t)}{dt} = -\left(\frac{\theta(t) + \hat{R}(\rho)}{L(\theta(t))} - \alpha\right).$$

Consider first the exact target ODE

$$\frac{d\theta(t)}{dt} = -\left(\frac{\theta(t) + R(\rho)}{L(\theta(t))} - \alpha\right) = -\frac{I(\theta(t))}{L(\theta(t))} + \alpha.$$

By definition of the S regime, there exists a unique θ^* such that $I(\theta^*) = \alpha L(\theta^*)$, which is the optimal control value. To verify that the ODE converges to this value, define the Lyapounov function $V(t) = \frac{1}{2}(I(\theta(t))/L(\theta(t)) - \alpha)^2$. Then

$$\frac{d}{dt}V(t) = \left(\frac{I(\theta(t))}{L(\theta(t))} - \alpha\right)\left(\frac{I(\theta(t))}{L(\theta(t))}\right)' \frac{d\theta(t)}{dt} = -\left(\frac{I(\theta(t))}{L(\theta(t))} - \alpha\right)^2 \left(\frac{I(\theta(t))}{L(\theta(t))}\right)' \leq 0,$$

showing that $V(t)$ is non-negative and the drift is strictly negative ($V'(t) < 0$) unless $\theta(t) = \theta^*$, meaning that $V(t) \to 0$, implying that $\theta(t) \to \theta^*$. The result now follows considering a small perturbation of the ODE:

$$\frac{d\tilde{\theta}(t)}{dt} = -\frac{I(\tilde{\theta}(t)) + R(\rho) + \beta)}{L(\tilde{\theta}(t))} - \alpha,$$

with corresponding limit point $\bar{\theta}$. Direct algebra provides the expression for $\bar{\theta} - \theta^*$ using:

$$\alpha = \frac{\bar{\theta} + \hat{R}}{c_1 + c_2\bar{\theta}} = \frac{\theta^* + R}{c_1 + c_2\theta^*},$$

which yields $\bar{\theta} - \theta^* = C\beta$. Using now that $\beta \approx$ Normal$(0, S^2/N)$ establishes the result. □

4 NON-STATIONARY MODEL

4.1 Daily Patterns for a CitiBike Station

In reality, the demand and usage of a public transportation system is not homogeneous in time. It is customary in transportation research to use segments of the day (classified as "peak" and "off-peak") that are defined by office administrators, who then use these for planning and pricing. If the day segments are pre-defined (say 0:00 – 6:30, 6:30 – 9:30, 9:30 – 12:00, 12:00 – 14:00, 14:00 – 16:30, 16:30 – 18:00, 18:00 - 24:00) then it is straightforward to estimate the corresponding demand rates (assuming corrections from historical data have been done). What we have discovered by looking at sampled data (by hand) is that stations do not exhibit the same behavior and segments are not necessarily constant throughout New York.

In order to create the input for the simulations, we need the matrix of demand rates $\{\lambda_{o,d}(t)\}$ as a function of the time of day. The Citibike website provides free access to the hourly number of bikes taken from each station (called the "departures"), and the hourly number of bikes that were docked at each station (called the "arrivals"). Figure 3 shows these numbers in the plots on top.

We implemented a simple change detection algorithm (Fournier and Vigneron 2011) that takes these historical hourly rates and estimates day segments and corresponding piecewise constant rates, shown in Figure 3 on the bottom. Our estimates consider the aggregated total number of customer arrivals during a period of nine months, excluding Summer and weekend days.

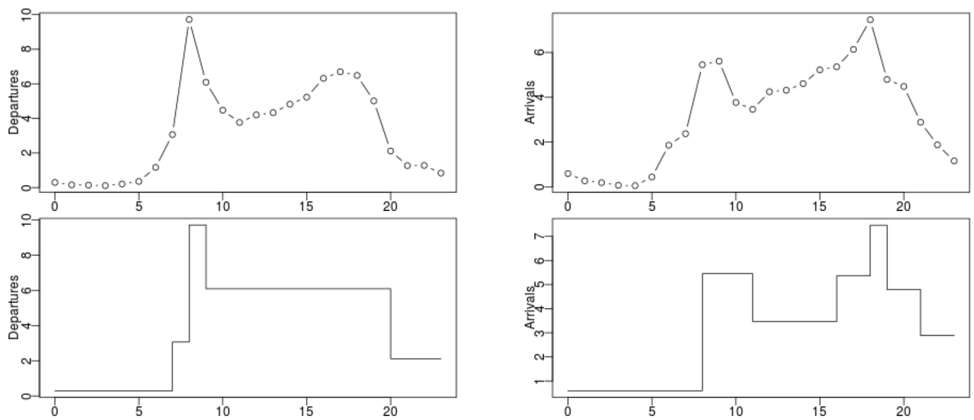

Figure 3: Automatic segment detection and demand rate estimation for station 72 (W 52 St & 11 Ave). The plots on top are the actual observed number of people arriving at the station to take bikes (Departures) or to park bikes (Arrivals). Below is the resulting step functions for day segments.

Single Station Non-Stationary Model. The model for the station occupation is a piecewise homogeneous B&D process with rates $\lambda(t), \mu(t)$ that are constant within each of the day segments. The segments of the day and rates are calculated from historical data for purposes of experimentation.

4.2 Implementing Stationary Policies

Table 1 summarizes the birth and death rates for a typical very busy station with several day segments. In addition to the corresponding values of λ and μ, we show the type of regime and the corresponding optimal policy obtained using the numerical methods described in Section 3: when the regime is super-starvation (SS) the optimal control (an integer) represents the minimum number of bikes before the truck is called. When the regime is starvation (S) the optimal control (a real number) is the waiting time policy before the truck is called to refill the station. For the filling and super-filling regimes we used symmetry in order to obtain the optimal threshold policies for when to call the truck (and in these cases it will take away bikes rather than filling the station). The neutral regime requires no intervention.

Table 1: Data for the simulation, $C = 58, \rho = 20$ and the target rate is $\alpha = 0.25$.

hour	λ	μ	θ^*	regime	hour	λ	μ	θ^*	regime
0-7	5	3	10.14	F	17-19	5	5	–	N
7-11	7	10	20.60	S	19-20	4	8	25	SS
11-16	10	6	0.99	F	20-21	10	7	20.54	F
16-17	4	4	–	N	21-24	2	3	46.80	S

We built a discrete event simulation for the experiments. Modern programming languages take advantage of parallelization in order to generate random variables (which is usually the most time consuming part of the execution). Thus, instead of generating the exponential clocks for the B&D events one at a time as the simulation evolves, we generate all of these times in advance. For the k-th segment on $[t_k, t_{k+1})$, the rates are λ_k, μ_k and the type of segment and optimal control is read from the table. The pre-calculation of the times of B&D events is done now using the conditional distribution of Poisson times given the number of events. So, first we generate $N \sim \text{Poisson}((\lambda_k + \mu_k)(t_{k+1} - t_k))$, then we generate N iid uniform random variables on $[t_k, t_{k+1})$. Finally, we sort these values and insert them in a list of events of "type 1". The numbers $\{t_k\}$ go directly in a list for events of "type 2" (change of regime). The list of events of "type 3" (truck arrival) is initialized at ∞. When the simulator looks for the next event, it compares the three current values of the lists. If the event is of type 2, the values of λ_k, μ_k, type of regime and the corresponding θ_k^* are updated. When the event is of type 2, a Bernoulli$(\lambda_k/(\lambda_k + \mu_k))$ is used to determine if it is a birth (1) or a death(0), it updates the state and other variables required for the control policy, and if the threshold is met, then the entry for the list of events of type 3 is updated with the current time plus ρ. When the event is a truck arrival it depends on the regime whether the station is replenished or emptied of bikes.

We ran $N = 1000$ replications of a day long simulation in order to estimate the dissatisfaction rate and cost of redistribution. We compute the total number of attempted rentals (deaths of the B&D model) and the number of *failed rentals* (those who find the station empty). Similarly we computed the total number of bike arrival (births in the model) and the number of *failed parkings*. The ratios provide the corresponding observed dissatisfaction indices (in percent values) D_r and D_p. The result yields $D_r = 11.1 \pm 0.0005$, and $D_p = 19.7 \pm 0.0006$. The total number of dissatisfied people (regardless of intended use of the station) divided by the total number of users (both clients and bikes arriving) was 15.65 ± 0.041. These results require some explanation. Recall that our optimal control is designed to achieve a target dissatisfaction rate of $\alpha = 0.25$. What we observe is that segments that will have large values of failed rentals are followed by regimes with opposite trend. Therefore calculating overall percentages is misleading. Table 2 summarizes the detailed results by segments of the day, and provides the corresponding estimates of the dissatisfaction indices by segment. These results seem to indicate very good performance, except for day segment 20-21, where the dissatisfaction rate is very low. We conjecture that the rate of events $\lambda + \mu$ in minutes is very fast, so that most of the segments have large enough duration to approximate the stationary operation. In Section 3 assume that the processes are homogeneous in time and they are calculated based on the long term (stationary) distribution. In contrast, if a station has regimes with mush less frequency of use $\lambda, \mu < 1$ it is very likely that the corresponding optimal controls from Section 3 will no longer be good (as seems to be the case with the day segment during 20-21 hours). This is the motivation for our model in Section 5, where we propose a different approach for the control policy that makes use of the streaming data as the system evolves.

Table 2: Results of simulating 1000 days using the policies in Table 1, with $\rho = 20$ minutes.

hours	truck calls	failed rentals	failed parkings	Tot clients (r)	Tot bikes (p)	α_r	α_p
0-7	5.14±0.02	7.88±0.29	513.64±3.02	1261.63±2.28	2101.14±2.84	0.006	0.244
7-11	2.24±0.03	572.87±3.37	6.36±0.31	2400.61±2.92	1678.15±2.50	0.239	0.004
11-16	7.76±0.03	12.45±0.35	741.08±3.47	1799.63±2.65	3001.26±3.43	0.007	0.247
16-17	0.51±0.03	1.52±0.34	11.21±0.83	238.97±0.98	239.36±0.95	0.006	0.047
17-19	0.00±0.00	7.49±0.90	13.93±1.16	599.06±1.54	600.21±1.51	0.012	0.023
19-20	1.98±0.01	116.71±1.44	1.97±0.12	478.68±1.32	240.47±0.94	0.244	0.008
20-21	0.67±0.03	1.87±0.17	115.49±2.90	418.91±1.25	600.16±1.50	0.004	0.192
21-24	0.29±0.03	125.49±1.69	1.31±0.14	539.26±1.36	359.98±1.16	0.233	0.004

5 STREAMING DATA: THE CALCULATOR

Our ultimate goal is to integrate streaming data and demand forecasts into the decision to trigger redistribution. Under what we believe to be a realistic model for data gathering in the near future, we suppose that customers use apps to request service and that they provide information about their desired destinations. In addition, estimated travel times are updated frequently by GPS, so at any given station, there is a good amount of information about future arrival of bikes that the birth-death process model above does not account for.To illustrate the incorporation of real-time data into model-based approaches, call the given station d and, for simplicity, assume that the travel times $T_{o,d}$ and time until redistribution ρ are constant.

5.1 Calculator

We describe now a "calculator" that will help to estimate the expected idle time and full time from a given start time (for example, the last truck arrival, or the beginning of the day) until an end time (for example, the next truck arrival if the truck was to be called immediately, or the start of the next day segment). The calculator uses known information and creates the unknown data via simulations, thus making our approach a "simulation analytics" approach.

Under our model for future bikeshare operation, each origin o communicates to the destination stations the scheduled arrival times of bikes (and these are updated frequently on route). Call $\mathscr{L}(t,t+\Delta)$ the ordered list of the n (known) scheduled arrival times of bikes from time t to $t+\Delta$, $\mathscr{L}(t,t+\Delta) = \{\tau_j\colon t \le \tau_j \le t+\Delta; 1 \le j \le n\}$, and set $\tau_0 = t, \tau_{n+1} = t+\Delta$. Finally, let $\tau_{\max}(o) \in \mathscr{L}(t,t+\Delta)$ denote the last scheduled (known) arrival from origin o, and for other origins o that do not have any scheduled arrivals and such that $T_{o,d} \le \Delta$ set $\tau_{\max}(o) = T_{o,d}$ as the initial shift for these new bike arrivals.

We will illustrate the idea assuming that the period $[t,t+\Delta)$ is completely contained in one day segment with constant rates λ,μ. The modification of the discrete event simulator when there are changes in day segments (regimes) are straightforward. Given $\mathscr{L}(t,t+\Delta)$, for every $0 \le j \le n$, the process behaves as a B&D process on $[\tau_j,\tau_{j+1})$ with death rate μ and birth rate

$$\lambda^{(j)} = \sum_{o\colon\ \tau_{\max}(o) \le \tau_j} \lambda_{o,d}.$$

This follows because for origins o such that $\tau_{\max}(o) > \tau_j$ there cannot be any unknown arrival of a bike from o to station d during the current time interval $[\tau_j,\tau_{j+1})$, under the assumption of constant travel times. In addition, the next subinterval will have initial occupation $X(\tau_{j+1}) = X(\tau_{j+1}^-) + \mathbf{1}_{\{X(\tau_{j+1}^-)<C\}}$. Finally, if there is a truck already scheduled to arrive in $[t,t+\Delta)$, this event is added to the simulation event list, and bikes are either taken out or put in the station, depending on the regime type.

When the calculator is called at time t, it performs several replications of the simulation and returns an estimator of the expected idle time $\mathbb{E}[\zeta(t,t+\Delta)]$ and the expected full time $\mathbb{E}[\varphi(t,t+\Delta)]$ over the time interval $[t,t+\Delta)$. The simulation model for the calculator is a hybrid discrete event/standard clock model.

5.2 Data-driven Control

The optimization problem that seeks to minimize the cost of redistribution subject to a quality of service (a bound on the dissatisfaction index) is motivated by an economic function. The reason why the dissatisfaction index is important for the bike share transportation industry is, ultimately, that dissatisfied customers are more likely not to renew their membership. An economic study of the system is outside the scope of this paper, but it is part of our research program. With this view, it is perhaps not appropriate to count the overall fraction of people that could not find bikes or docks, over the whole day. This is because during a peak hour segment the number of dissatisfied customers may have a larger impact on the long term economics. This fraction is artificially counterbalanced when considering also the periods of little usage of the system, where most users will be satisfied.

In any case, the data-driven control is a flexible tool that can handle any of the models for the dissatisfaction index, be it by day segment, by specific stations, by customer type, etc. We illustrate this assuming that from the start of the k-th day segment (t_k) it is desired to keep at most a fraction α of dissatisfied customers for the segment $[t_k,t_{k+1})$.

For any $t_1 < t_2$ define the following counters:

$$d_r(t_1,t_2): \text{ number of failed rentals on } [t_1,t_2)$$
$$d_p(t_1,t_2): \text{ number of failed parkings on } [t_1,t_2)$$
$$a_r(t_1,t_2): \text{ number of arriving customers on } [t_1,t_2)$$
$$a_p(t_1,t_2): \text{ number of arriving bikes on } [t_1,t_2).$$

Suppose that the current regime is such that $\lambda < \mu$, so the dissatisfaction index counts failed rentals. The calculator is called at time $t > t_k$ using $\Delta = \rho$ to estimate the expected dissatisfaction until the next truck arrival, if the call for redistribution is to be made "now" (at time t). When $\rho > t_k - t$ the end of the current regime will happen before any redistribution can be made, so we will call the calculator only if the following regime also has $\lambda < \mu$. Otherwise the next segment may have a different dissatisfaction criterion (failed parkings) so we do not call the calculator until t_{k+1}.

When the calculator is called at time $t \in (t_k, t_{k+1})$ the fraction of the expected failed rentals is

$$f(t) = \frac{d_r(t_k,t) + \mathbb{E}[\zeta(t,t+\Delta)]}{a_r(t_k,t) + \mu\Delta},$$

using conditional expectations given the information up to time t. One possible control rule is to call the truck if $f(t) \geq \alpha$. The calculator is called again every time that there is an event, be it an arriving customer, an arriving bike, a truck arrival, or a change in regime (day segment).

6 FUTURE RESEARCH

For the streaming data, we programmed the calculator using a discrete-event simulator, but the data manipulation required makes it very slow to test. As part of the research plan, we will investigate techniques to improve the efficiency of the simulator, as well as seek to find analytical solutions or approximations. This will require careful theoretical analysis to merge solutions of the backward/forward Kolmogorov equations (Ross 2014) with deterministic data. To our knowledge, such a mixed approach is new (though one could view it as a hybrid system) and will contribute to the emerging area of simulation analytics. This paper deals with a single station view of the problem, so optimal controls are defined under a selfish viewpoint. The control policies are used to trigger alarms in all stations, and truck scheduling will have to use this multi-dimensional triggers to best accommodate the demand, which is part of our current research, but outside the scope of this paper.

ACKNOWLEDGMENTS

The authors would like to acknowledge the work of Cynthiaann Bryant who did data analysis on the CitiBike historical data and Ben Elias Morgenroth, who worked on some preliminary code. This work was partially supported by the CUNY Institute for Computer Simulation, Stochastic Modeling and Optimization.

REFERENCES

Bruglieri, M., A. Colorni, and A. Luè. 2014. "The vehicle relocation problem for the one-way electric vehicle sharing: an application to the Milan case". *Procedia-Social and Behavioral Sciences* 111:18–27.

Bullock, C., F. Brereton, and S. Bailey. 2017. "The economic contribution of public bike-share to the sustainability and efficient functioning of cities". *Sustainable Cities and Society* 28:76–87.

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

Chemla, D., F. Meunier, T. Pradeau, R. W. Calvo, and H. Yahiaoui. 2013. "Self-service bike sharing systems: simulation, repositioning, pricing". Technical Report hal-00824078, Centre dEnseignement et de Recherche en Mathmatiques et Calcul Scientifique (CERMICS).

Fournier, H., and A. Vigneron. 2011. "Fitting a step function to a point set". *Algorithmica* 60 (1): 95–109.

NYCDP 2009. "Bike-Share Opportunities in New York City".

Raviv, T., M. Tzur, and 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.

Ross, S. M. 2014. *Introduction to probability models*. Academic Press.

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

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

Singhvi, D., S. Singhvi, P. I. Frazier, S. G. Henderson, E. O'Mahony, D. B. Shmoys, and D. B. Woodard. 2015. "Predicting Bike Usage for New York City's Bike Sharing System.". In *AAAI Workshop: Computational Sustainability*.

Taylor, and Karlin. 1998. *An Introduction to stochastic modeling*. Academic Press.

Vázquez-Abad, F., and B. Heidergott. 2017. *Gradient Based Stochastic Optimization*. Lecture Notes, to be published.

Vázquez-Abad, F. J., and L. Fenn. 2016. "Mixed optimization for constrained resource allocation, an application to a local bus service". In *Proceedings of the 2016 Winter Simulation Conference*, edited by T. M. K. Roeder, P. I. Frazier, R. Szechtman, E. Zhou, T. Huschka, and S. E. Chick, 871–882. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

AUTHOR BIOGRAPHIES

FELISA J. VÁZQUEZ-ABAD is Professor of Computer Science at Hunter College of the City University New York (CUNY). She is Executive Director of the CUNY Institute for Computer Simulation, Stochastic Modeling and Optimization that she helped to create in 2013. She has a Ph.D. in Applied Mathematics from Brown University. She was a professor at the University of Montreal, Canada in 1993 until 2004 when she became a professor at the University of Melbourne, Australia, until 2009. Her interests focus on the optimization of complex systems under uncertainty, primarily to build efficient self- regulated learning systems. She has applied novel techniques for simulation and optimization in telecommunications, transportation, finance and insurance and she is interested by real life problems. Her email address is felisav@hunter.cuny.edu.

MICHAEL C. FU holds the Smith Chair of Management Science in the Smith School of Business, University of Maryland, College Park. He served as 2011 WSC Program Chair and is a Fellow of INFORMS and IEEE. His e-mail address is mfu@umd.edu.

SILVANO BERNABEL is a research assistant at the CUNY Institute for Computer Simulation, Stochastic Modeling and Optimization. He holds a M.A. in Pure Mathematics from Hunter College and a B.E. in Electrical Engineering from City College CUNY. His area of interest is in Data Science. His email address is sb1037@hunter.cuny.edu.