Under the Hood of Bike-sharing

Shane Henderson
Cornell University

Based on joint work with
David Shmoys
Daniel Freund
Nanjing Jian
Eoin O'Mahony

And Hangil Chung, Ashkan Norouzi Fard, Aaron Ferber,
Alice Paul, Holly Wiberg, & David Williamson

Supported in part by grants from the National Science Foundation

Background on Motivate & Citi Bike

- Motivate operates systems in NYC, Chicago, Boston, Bay Area, Washington...
- Across systems, more than 100M rides since 2010

Citi Bike	
Stations	750+
Docks	25k
Bikes	12k
Subscribers	147k
Rides in ‘18	17.5M
Record day (6/26/2018)	80,624

Allocating bikes

$$\text{minimize}_b \sum_i c_i(b_i)$$

$$\text{s.t.} \sum_i b_i \leq B$$

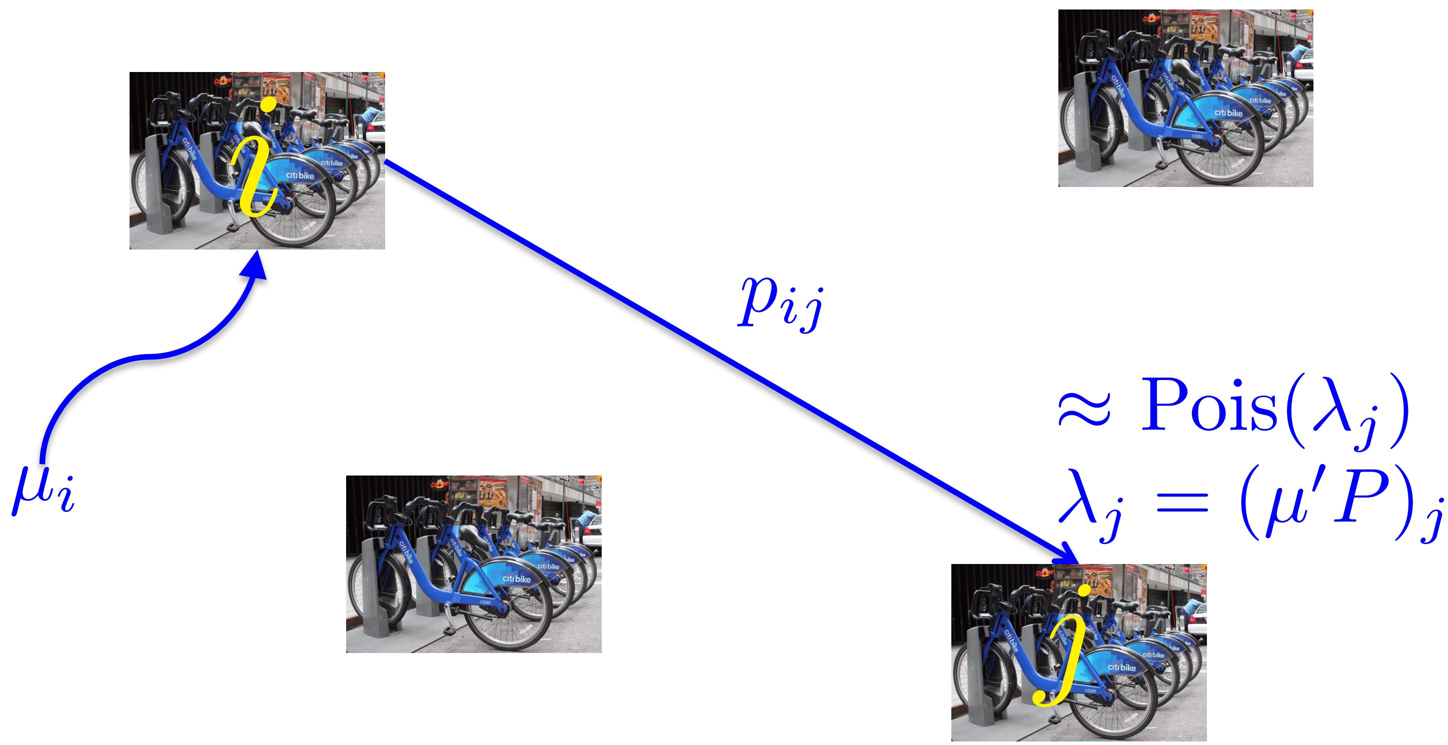
$$\forall i : 0 \leq b_i \leq r_i, \text{integer}$$

Choice of Objective Function

- Subscription system
- Minimize user dissatisfaction
- View planning period as 6am-midnight
- Objective: expected number of bike & dock stock-outs (= # upset customers) in a day

- Challenging due to tidal flows of commuters
- For now: Ignore time dependence of flow rates

The “Physics”



Using fact that delayed Poisson process is Poisson process

The System Decomposes

- Assume never run out of bikes “upstream” (strong)
- Then for any station
    - Biker arrival process is Poisson
    - Biker return process is Poisson
    - And they’re independent, assuming “self loops” are negligible
- Each station individually behaves like an $M_t/M_t/1/N$ queue
- Huge simplification! Treat each station alone, not a network

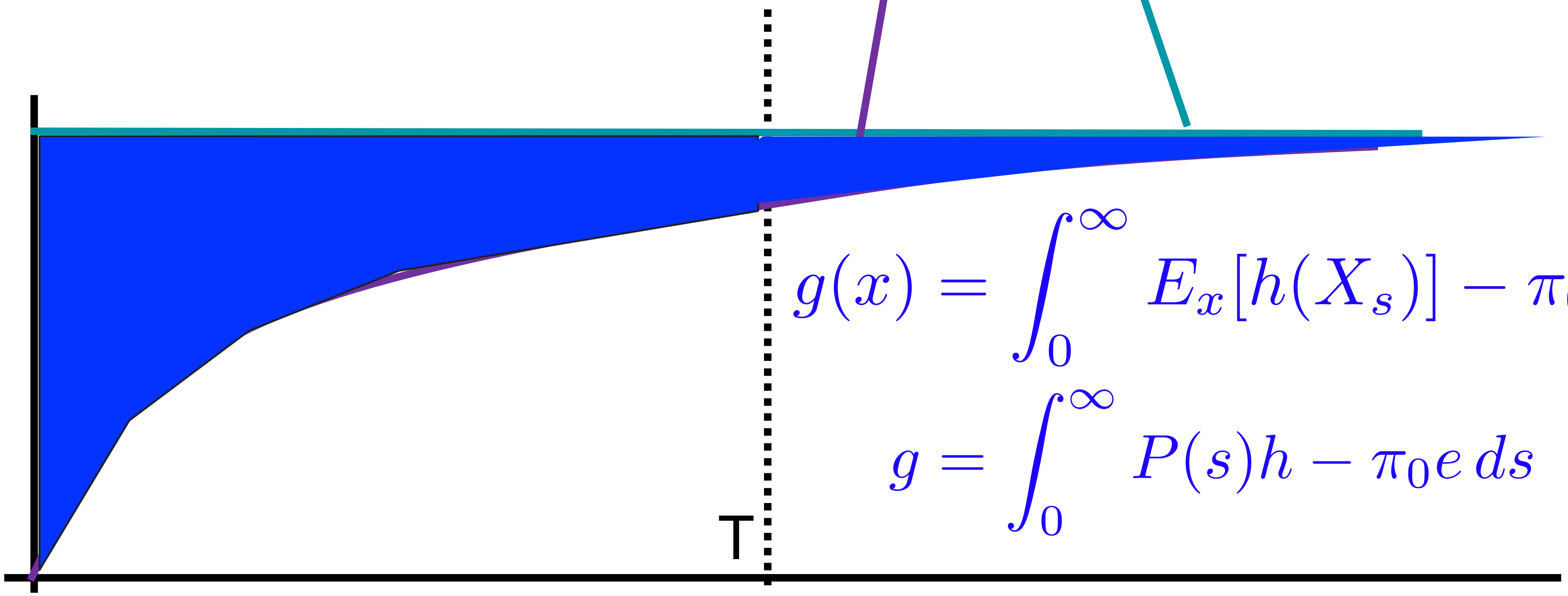
Computation for each station

- Break down the day into 30 minute intervals
- Assume rates are piecewise constant over intervals
- Compute objective function (how?) in each interval and add them up
- In each interval the number of bikes as a function of time is a continuous-time Markov chain
- Rate matrix A
- Different rate matrices in different intervals
- Expected number of bike outages = arrival rate * time empty
  = arrival rate * (30 minutes * fraction of time station is empty)
- $\boldsymbol{\pi}A=0$ and then look at $\boldsymbol{\pi}_0$?

- Model one time interval, [0, 30] = [0, T]
- Define h(x) = I(x=0) (similar approach for upper boundary)

$$v(x) = E_x \int_0^T h(X_s)\, ds$$

$$w(x) = \pi_0 T + \int_0^T \boxed{E_x[h(X_s)]} - \pi_0 \, ds$$



$$g(x) = \int_0^\infty E_x[h(X_s)] - \pi_0 \, ds$$

$$g = \int_0^\infty P(s)h - \pi_0 e \, ds$$

- Kolmogorov's backward equation P'(t) = A P(t)

$$g = \int_0^\infty P(s)h - \pi_0 e \, ds = \int_0^\infty P(s)(h - \pi_0 e) \, ds$$

$$\begin{aligned} Ag &= \int_0^\infty AP(s)h_c \, ds \\ &= \int_0^\infty P'(s)h_c \, ds \\ &= [P(s)h_c]_0^\infty \\ &= -h_c \end{aligned}$$

From Poisson to what we want

$$
\begin{aligned}
v &= \pi_0 e T + \int_0^T P(s) h_c \, ds \\
&= \pi_0 e T + \int_0^\infty P(s) h_c \, ds - \int_T^\infty P(s) h_c \, ds \\
&= \pi_0 e T + g - P(T) \int_0^\infty P(s) h_c \, ds \\
&= \pi_0 e T + (I - P(T)) g
\end{aligned}
$$

- P'(t) = A P(t). So P(t) = e^{At}

Piecing things together

$$v = \ldots$$

$$p(30) = p(0)P$$

$$v = \ldots$$

$$p(60) = p(30)P$$

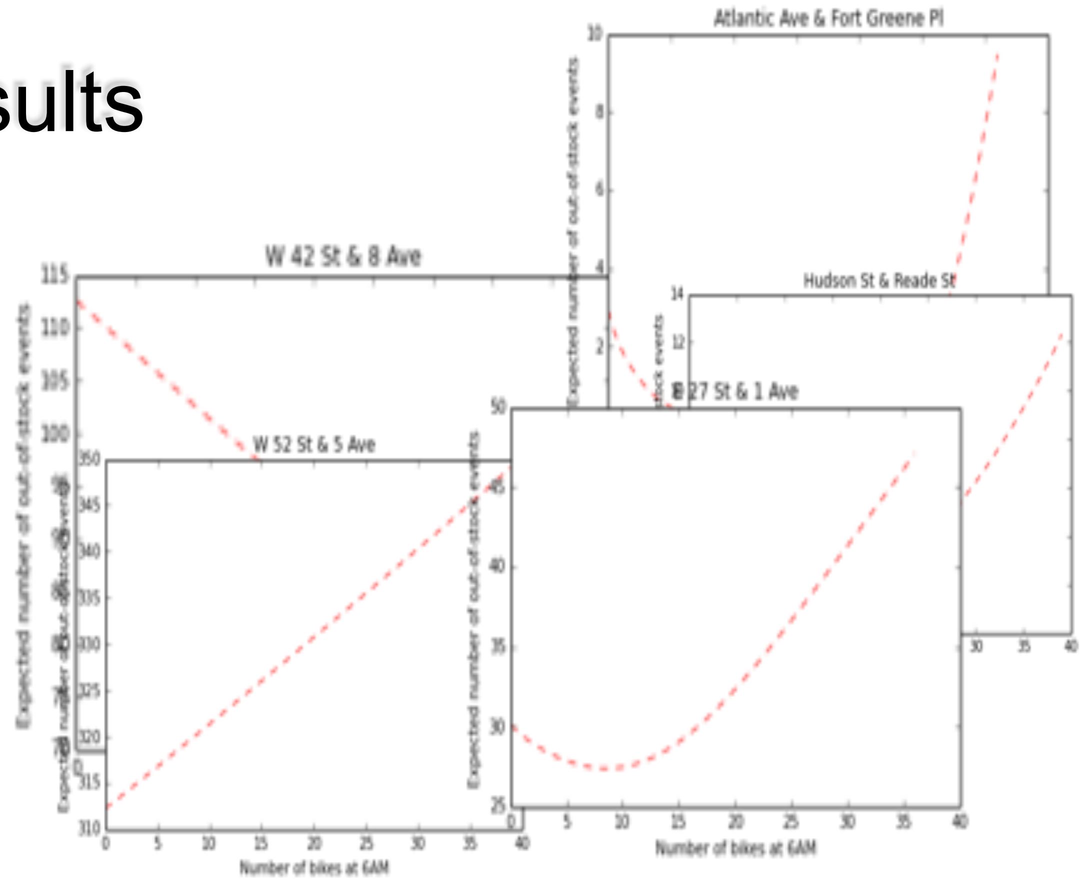
$$\ldots$$

30

60

90

Results

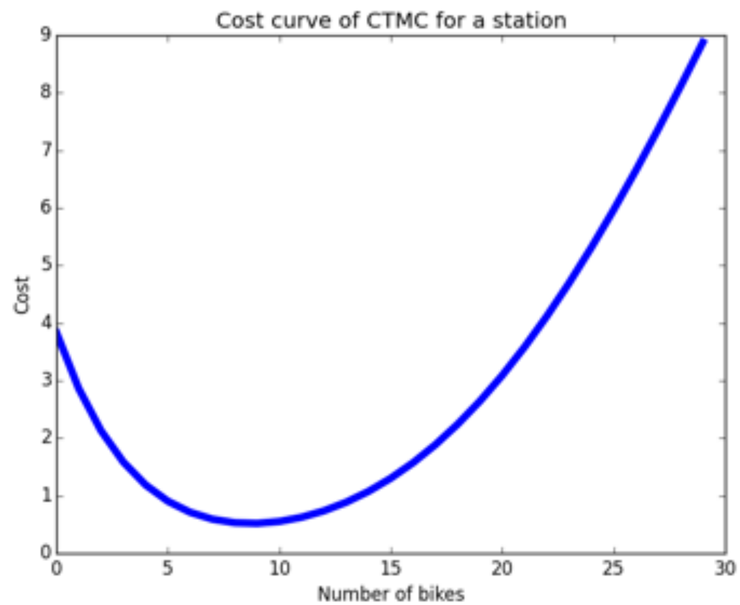


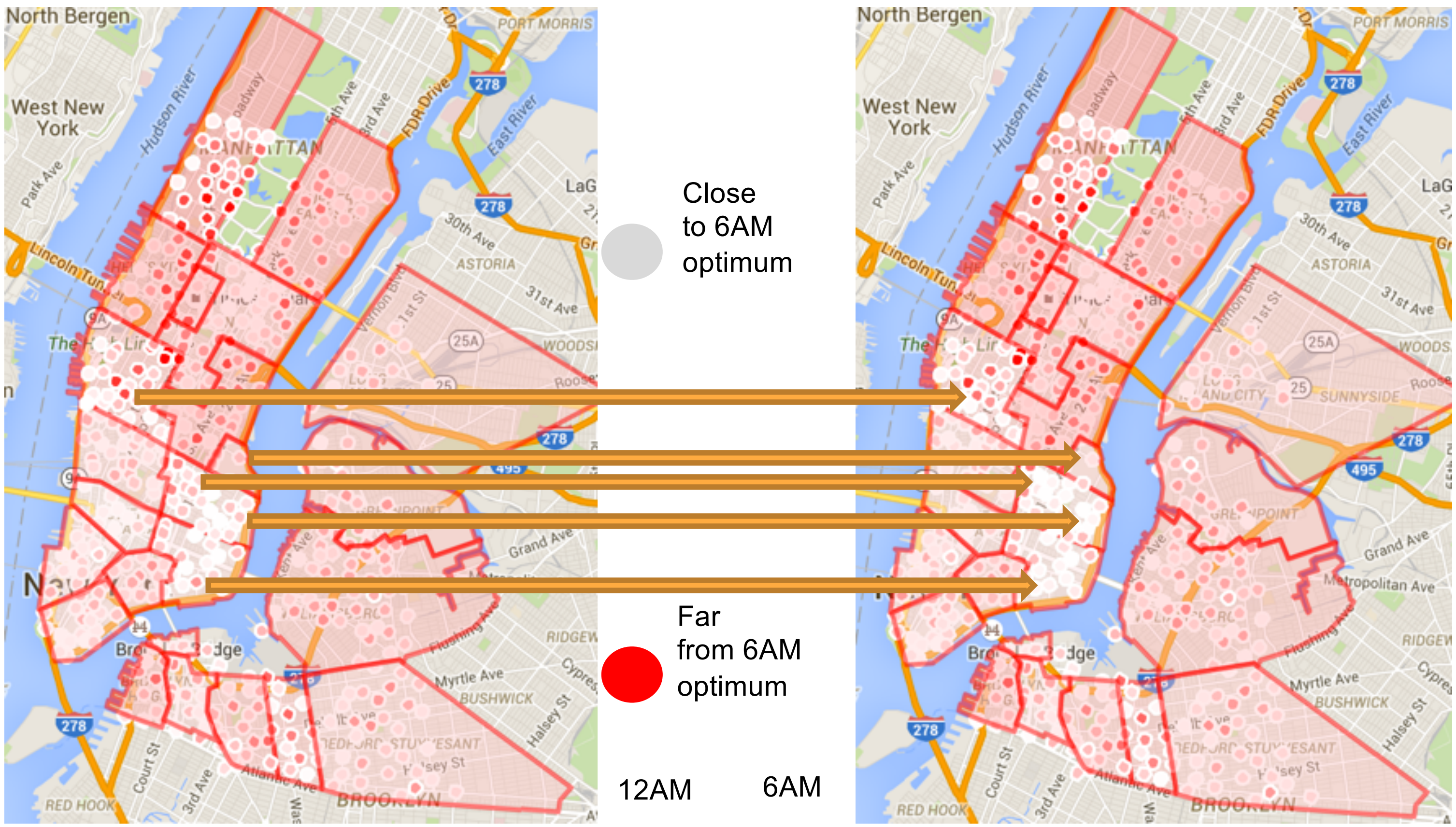
Atlantic Ave & Fort Greene Pl
Hudson St & Reade St
W 42 St & 8 Ave
W 52 St & 5 Ave
E 27 St & 1 Ave
Expected number of out-of-stock events
Number of bikes at 6AM

Daily Planning Problem: Optimization

Place bikes to minimize E[#upset cust]

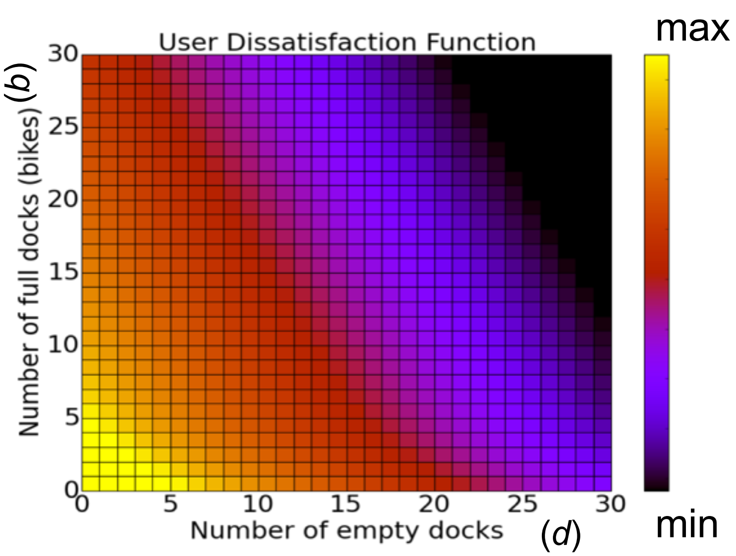
Thm: Cost function convex in bikes





Need to go beyond moving bikes overnight. Move docks?

Optimization Model



Out-of-stock events at station i when initialized with d_i empty docks and b_i bikes (full docks)

$$\text{minimize}_{(\vec{d},\vec{b})} \quad \sum_i c_i(d_i, b_i)$$

$$s.t.: \quad \sum_i (d_i + b_i) \leq D + B$$ a budget on docks,

$$\sum_i b_i \leq B$$ a budget on bikes,

$$\sum_i |\bar{d}_i + \bar{b}_i - d_i - b_i| \leq 2z$$ a budget on docks moved,

$$\forall i: \; l_i \leq d_i + b_i \leq u_i$$

and bounds on the size of each station

Multimodularity

"multidimensional diminishing returns"

A function c is multimodular if

1. $c(d+1,b) - c(d+2,b) \leq c(d,b) - c(d+1,b)$
2. $c(d+1,b) - c(d+1,b+1) \leq c(d,b) - c(d,b+1)$
3. $c(d,b-1) - c(d+1,b-1) \leq c(d-1,b) - c(d,b)$

Proposition: The cost-function at each station is multimodular.
Proof by pathwise induction on sequence of events

Based on August 2016 data

Potential Improvements

Allocation	Cost in long run (no rebalancing)
Long-run optimal dock-allocation	18349 (OPT)
Allocation currently in place	23851 (+30.0%)

Overnight Rebalancing

Mid-Rush Rebalancing

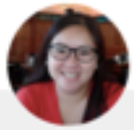
Corral Placement

Bike Angels

Featured Bike Angels

Hans, Midtown

"I like to do a 'Bike Angels workout' where I'll bike north in Central Park to an empty station, then run back down the park and repeat a few times."



Kelly, Lower East Side

"I'm not really a counter-commuter but I tend to check the Bike Angels map before each of my Citi Bike trips, and I earn a lot of points that way."

Bike Angels are Citi Bike riders that...

improve the availability of bikes and docks for fellow riders and earn rewards along the way

Sign Up

How to earn points

You earn Bike Angel points depending on the pair of stations involved ... your Citi Bike trips. See below for examples of how **Pick Up and Drop Off points** work!

1 point

Start at neutral station, bike to **1-point Drop Off**

2 points

Start at **2-point Pick Up** station, bike to neutral station

3 points

Start at **2-point Pick Up** station, bike to **1-point Drop Off**


Score: 49
LEADERBOARD
APRIL 2
3:24 PM
ANGEL
POINTS
1 JM 276
2 WF 101
3 FH 83
4 BH 56
5 GR 33
6 TH 30
7 VH 30
8 CS 30
9 AA 29
10 MM 29

Simulation for System Design

- Why:
  - Simulation better captures the “interactions”
  - Allows the modeling of more complex customer behaviors and (mid-day) rebalancing efforts
- Simulation yields objective function at one dock and bike allocation. How to optimize the allocation?
- Random search variants unlikely to be useful
- Want “∇c”
- Finite differences too expensive
  - With 750 stations, 1501 simulations

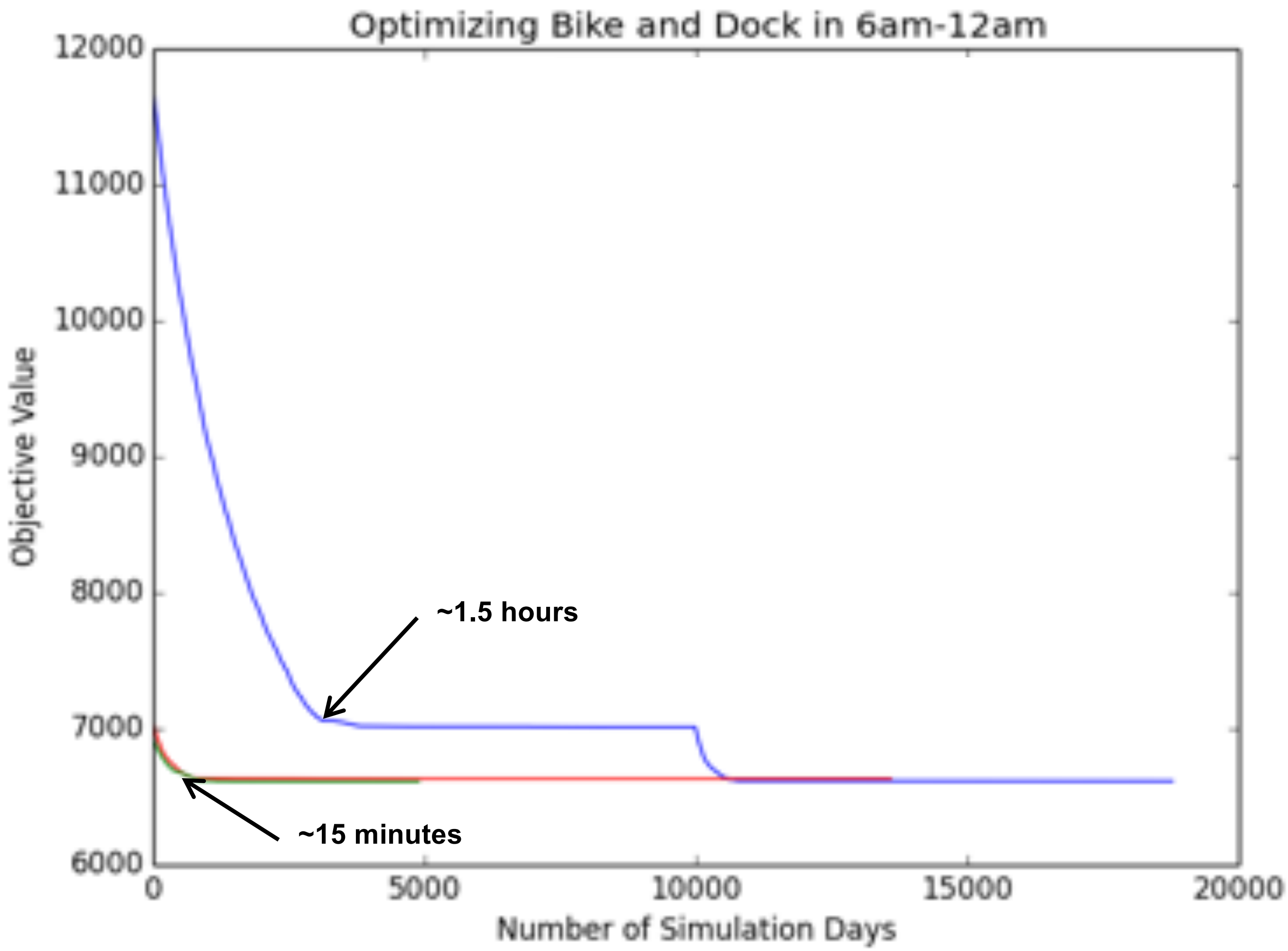
White Box: Perturbation Analysis

- Don't treat simulation model as a black box
- Track objective C, but also objective with one extra dock, one extra bike at each station (so nominal + 750 stations * 2 = 1501 values)
- Compute differences as ∇C
- With replications we estimate $E(\nabla C)$
- What we really want is $\nabla c = \nabla E(C)$
- Does $E(\nabla C) = \nabla E(C)$?
- This is the central question in perturbation analysis methods for gradient estimation

But does $E(\nabla C) = \nabla E(C)$?

- Nope
  - Gradients we estimate point in the wrong direction
- But do they provide a descent direction when far from the optimal solution?
- Let’s try it, optimizing over bikes only

Optimizing over bikes



Simulation Commentary

- Estimated gradients are biased, point in wrong direction
- Still a descent direction when far from optimal.
- Can prove that under stylized settings, biased gradients can get you within a neighbourhood of an optimal solution
- Sufficient for many applications
- Perhaps too much emphasis on $E(\nabla C) = \nabla E(C)$

Conclusion

- ❖ Rebalancing is a major expenditure in bike-sharing
  - ➢ Reallocate capacity to require less rebalancing
  - ➢ Poisson's equation to compute transient performance
  - ➢ Multimodality
- ❖ Bike Angels originated at Cornell
- ❖ Simulation optimization: Embrace biased gradients?

Thank You!!