PUTTING DATA in the DRIVER'S SEAT

Optimizing Earnings for On-Demand Ride-Hailing

Harshal Chaudhari John Byers Evimaria Terzi

BOSTON UNIVERSITY

Problem Motivation

The Washington Post

Democracy Dies in Darkness

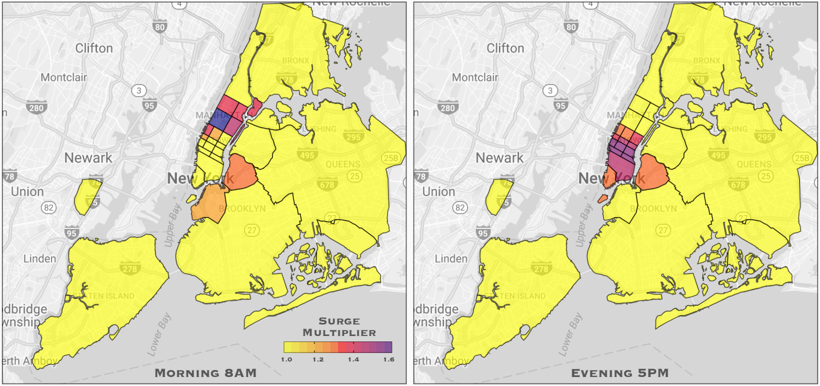
The Switch

How much Uber drivers actually make per hour

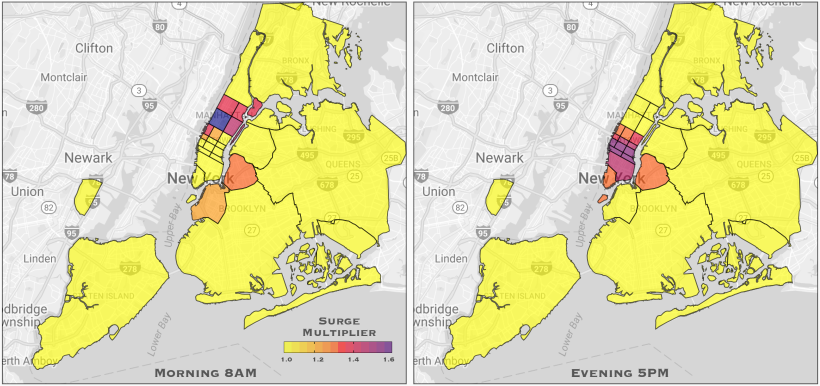
By Jacob Bogage June 27, 2016 Email the author

‘ … about half of the hours logged into the app in New York City are spent without passengers in the car. And that’s in one of Uber’s busiest markets in the country. ’

Spatio-Temporal Demand Variation in NYC



Spatio-Temporal Demand Variation in NYC



As an Uber driver, how do you maximize earnings?

Silicon Valley Steps In ...

#1 Supporting Platform For On-Demand Workers

Get Access **SherpaShare** Suite
Dashboard + **Tracker** + **Pulse** + **Heatmap** + ...

SherpaShare provides ‘myopic’ guide to drivers in form of heat-maps. No predictions!

Literature

- Why is Uber an attractive platform for drivers?
  - M. K. Chen and Sheldon (2016) and Hall and Krueger (2016).

Literature

- Why is Uber an attractive platform for drivers?
  - M. K. Chen and Sheldon (2016) and Hall and Krueger (2016).

- Understanding dynamic pricing
  - L. Chen, Mislove, and Wilson (2015) did an empirical study on surge pricing in Uber.
  - Castillo, Knoepfle, and Weyl (2017) show that surge pricing is essential for success of ride-hailing platforms.
  - Banerjee, Johari, and Riquelme (2015), Ma, Fang, and Parkes (2018), and Bimpikis, Candogan, and Daniela (2016) provide theoretical models to study dynamic pricing.

Literature

- Why is Uber an attractive platform for drivers?
  - M. K. Chen and Sheldon (2016) and Hall and Krueger (2016).
- Understanding dynamic pricing
  - L. Chen, Mislove, and Wilson (2015) did an empirical study on surge pricing in Uber.
  - Castillo, Knoepfle, and Weyl (2017) show that surge pricing is essential for success of ride-hailing platforms.
  - Banerjee, Johari, and Riquelme (2015), Ma, Fang, and Parkes (2018), and Bimpikis, Candogan, and Daniela (2016) provide theoretical models to study dynamic pricing.
- Taxi driver models
  - Camerer et al. (1997) study strategies of taxi drivers.
  - Zhang et al. (2017) devise models for optimal taxi dispatch.

Literature

- Why is Uber an attractive platform for drivers?
  - M. K. Chen and Sheldon (2016) and Hall and Krueger (2016).
- Understanding dynamic pricing
  - L. Chen, Mislove, and Wilson (2015) did an empirical study on surge pricing in Uber.
  - Castillo, Knoepfle, and Weyl (2017) show that surge pricing is essential for success of ride-hailing platforms.
  - Banerjee, Johari, and Riquelme (2015), Ma, Fang, and Parkes (2018), and Bimpikis, Candogan, and Daniela (2016) provide theoretical models to study dynamic pricing.
- Taxi driver models
  - Camerer et al. (1997) study strategies of taxi drivers.
  - Zhang et al. (2017) devise models for optimal taxi dispatch.

We focus on earnings of individual, self-interested drivers empowered by data.

Modeling a Strategic Driver

Modeling a City

- Discrete model:
  - **Nodes**: Set of n city-zones ($\mathcal{X}$).
  - **Edges**: Rides between each pair of zones.
  - **Time**: advances in discrete time steps.

Modeling a City

- Discrete model:
  - **Nodes**: Set of n city-zones ($\mathcal{X}$).
  - **Edges**: Rides between each pair of zones.
  - **Time**: advances in discrete time steps.
- Empirical Transition Matrix (F):
  - $f(i,j)$: probability of passenger in zone i traveling to zone j.
- Rewards Matrix (R):
  - $r(i,j)$: net reward from zone i to zone j.
  - $r(i,j)$ = earnings(i,j) - cost(i,j).
- Travel Time Matrix (T):
  - $\tau(i,j)$: travel time from zone i to zone j.

Modeling a City

- Discrete model:
  - **Nodes**: Set of n city-zones ($\mathcal{X}$).
  - **Edges**: Rides between each pair of zones.
  - **Time**: advances in discrete time steps.
- Empirical Transition Matrix (F):
  - $f(i, j)$: probability of passenger in zone i traveling to zone j.
- Rewards Matrix (R):
  - $r(i, j)$: net reward from zone i to zone j.
  - $r(i, j)$ = earnings(i, j) - cost(i, j).
- Travel Time Matrix (T):
  - $\tau(i, j)$: travel time from zone i to zone j.

In general, all these matrices are time-dependent, their entries change throughout the day.

Modeling a Driver

- Home Zone:
  - The zone from which driver starts the day.

Modeling a Driver

- **Home Zone:**
  - The zone from which driver starts the day.
- **Finite Horizon (N):**
  - Total time steps in optimization period.

Modeling a Driver

- **Home Zone:**
  - The zone from which driver starts the day.
- **Finite Horizon (N):**
  - Total time steps in optimization period.
- **Driving Budget (B):**
  - Maximum time steps a driver is willing to drive. Obviously, $B \leq N$.

Modeling a Driver

- **Home Zone:**
  - The zone from which driver starts the day.
- **Finite Horizon (N):**
  - Total time steps in optimization period.
- **Driving Budget (B):**
  - Maximum time steps a driver is willing to drive. Obviously, $B \leq N$.
- **Actions ($\mathcal{A}$):**
  - Get Passenger
  - Go Home
  - Relocate

Modeling a Driver

- Home Zone:
  - The zone from which driver starts the day.
- Finite Horizon (N):
  - Total time steps in optimization period.
- Driving Budget (B):
  - Maximum time steps a driver is willing to drive. Obviously, $B \leq N$.
- Actions ($\mathcal{A}$):
  - Get Passenger
  - Go Home
  - Relocate

Naive Uber Driver

Modeling a Driver

- Home Zone:
  - The zone from which driver starts the day.
- Finite Horizon (N):
  - Total time steps in optimization period.
- Driving Budget (B):
  - Maximum time steps a driver is willing to drive. Obviously, $B \leq N$.
- Actions ($\mathcal{A}$):
  - Get Passenger
  - Go Home
  - Relocate

  Flexible Strategy

Modeling a Driver

- Home Zone:
  - The zone from which driver starts the day.
- Finite Horizon (N):
  - Total time steps in optimization period.
- Driving Budget (B):
  - Maximum time steps a driver is willing to drive. Obviously, $B \leq N$.
- Actions ($\mathcal{A}$):
  - Get Passenger
  - Go Home
  - Relocate

Relocation Strategy

Modeling a Driver

- **Home Zone:**
  - The zone from which driver starts the day.
- **Finite Horizon (N):**
  - Total time steps in optimization period.
- **Driving Budget (B):**
  - Maximum time steps a driver is willing to drive. Obviously, $B \leq N$.
- **Actions ($\mathcal{A}$):**

$$\left.\begin{array}{l} \text{– Get Passenger} \\ \text{– Go Home} \\ \text{– Relocate} \end{array}\right\} \text{Flexible-Relocation Strategy}$$

Modeling a Driver

- **Home Zone:**
  - The zone from which driver starts the day.
- **Finite Horizon (N):**
  - Total time steps in optimization period.
- **Driving Budget (B):**
  - Maximum time steps a driver is willing to drive. Obviously, $B \leq N$.
- **Actions ($\mathcal{A}$):**
  - Get Passenger
  - Go Home
  - Relocate
- **Driver Policy (π):**
  - Time and location dependent actions takes by driver.
  - Let Π denote entire policy space of size $\left|n \times N \times B \times \mathcal{A}\right|$.

Problem Definition

MaxEarnings Problem: Given a set of time-evolving F, T and R matrices, as well as the driver's budget B, find a policy π^* such that:

$$\pi^* = \arg\max_{\pi \in \Pi} \mathcal{E}(\pi, F, T, R, B)$$

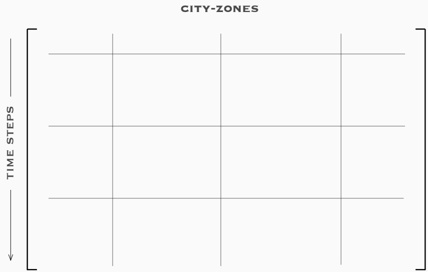
where $\mathcal{E}(.)$ denotes total expected earnings.

Problem Definition

MaxEarnings Problem: Given a set of time-evolving F, T and R matrices, as well as the driver's budget B, find a policy π^* such that:

$$\pi^* = \arg\max_{\pi \in \Pi} \mathcal{E}(\pi, F, T, R, B)$$

where $\mathcal{E}(.)$ denotes total expected earnings.



Problem Definition

MaxEarnings Problem: Given a set of time-evolving F, T and R matrices, as well as the driver's budget B, find a policy π^* such that:

$$\pi^* = \arg\max_{\pi \in \Pi} \mathcal{E}(\pi, F, T, R, B)$$

where $\mathcal{E}(.)$ denotes total expected earnings.

Dynamic Program solvable in $O(n^2 NB)$.

Problem Definition

MaxEarnings Problem: Given a set of time-evolving F, T and R matrices, as well as the driver's budget B, find a policy π^* such that:

$$\pi^* = \arg\max_{\pi \in \Pi} \mathcal{E}(\pi, F, T, R, B)$$

where $\mathcal{E}(.)$ denotes total expected earnings.

Dynamic Program solvable in $O(n^2 NB)$.

Problem Definition

MaxEarnings Problem: Given a set of time-evolving F, T and R matrices, as well as the driver's budget B, find a policy π^* such that:

$$\pi^* = \arg\max_{\pi \in \Pi} \mathcal{E}(\pi, F, T, R, B)$$

where $\mathcal{E}(.)$ denotes total expected earnings.

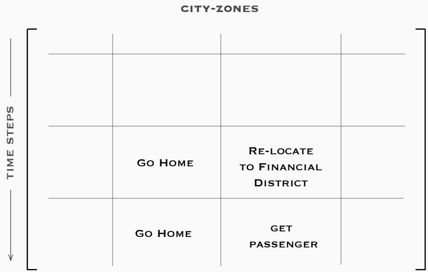
Dynamic Program solvable in $O(n^2NB)$.

Problem Definition

MaxEarnings Problem: Given a set of time-evolving F, T and R matrices, as well as the driver's budget B, find a policy π^* such that:

$$\pi^* = \arg\max_{\pi \in \Pi} \mathcal{E}(\pi, F, T, R, B)$$

where $\mathcal{E}(.)$ denotes total expected earnings.



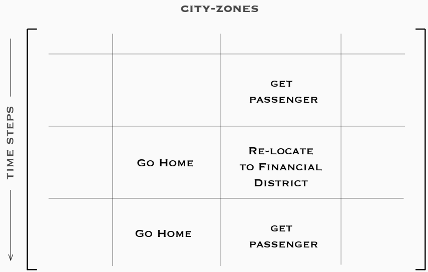
Dynamic Program solvable in $O(n^2NB)$.

Problem Definition

MaxEarnings Problem: Given a set of time-evolving F, T and R matrices, as well as the driver's budget B, find a policy π^* such that:

$$\pi^* = \arg\max_{\pi \in \Pi} \mathcal{E}(\pi, F, T, R, B)$$

where $\mathcal{E}(.)$ denotes total expected earnings.



Dynamic Program solvable in $O(n^2NB)$.

Problem Definition

MaxEarnings Problem: Given a set of time-evolving F, T and R matrices, as well as the driver's budget B, find a policy π^* such that:

$$\pi^* = \arg\max_{\pi \in \Pi} \mathcal{E}(\pi, F, T, R, B)$$

where $\mathcal{E}(.)$ denotes total expected earnings.

Dynamic Program solvable in $O(n^2NB)$.

Empirical Experiments

Data Sources

- NYC taxi rides dataset (2015-2016)
  - Over 200,000 taxi rides per day, 29 city-zones.
  - Pickup/ dropoff times and location co-ordinates.

Data Sources

- NYC taxi rides dataset (2015-2016)
  - Over 200,000 taxi rides per day, 29 city-zones.
  - Pickup/ dropoff times and location co-ordinates.

  Used to generate time-evolving empirical transition matrix (F)

Data Sources

- NYC taxi rides dataset (2015-2016)
  - Over 200,000 taxi rides per day, 29 city-zones.
  - Pickup/ dropoff times and location co-ordinates.

  Used to generate time-evolving empirical transition matrix (F)
- Uber Rides API
  - Provides estimates for price and travel time.
  - Surge multiplier.

Data Sources

- NYC taxi rides dataset (2015-2016)
  - Over 200,000 taxi rides per day, 29 city-zones.
  - Pickup/ dropoff times and location co-ordinates.

  Used to generate time-evolving empirical transition matrix (F)

- Uber Rides API
  - Provides estimates for price and travel time.
  - Surge multiplier.

  Used to generate rewards matrix (R) and travel time matrix (T)

Data Sources

- NYC taxi rides dataset (2015-2016)
  - Over 200,000 taxi rides per day, 29 city-zones.
  - Pickup/ dropoff times and location co-ordinates.

  Used to generate time-evolving empirical transition matrix (F)

- Uber Rides API
  - Provides estimates for price and travel time.
  - Surge multiplier.

  Used to generate rewards matrix (R) and travel time matrix (T)

- Methodology:

2016

1st Sunday of January
1 pm
Brooklyn to Williamsburg

NYC Taxi Ride

Data Sources

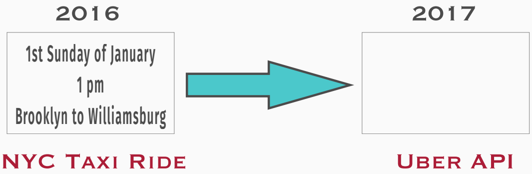
- NYC taxi rides dataset (2015-2016)
  - Over 200,000 taxi rides per day, 29 city-zones.
  - Pickup/ dropoff times and location co-ordinates.

  Used to generate time-evolving empirical transition matrix (F)

- Uber Rides API
  - Provides estimates for price and travel time.
  - Surge multiplier.

  Used to generate rewards matrix (R) and travel time matrix (T)

- Methodology:



Data Sources

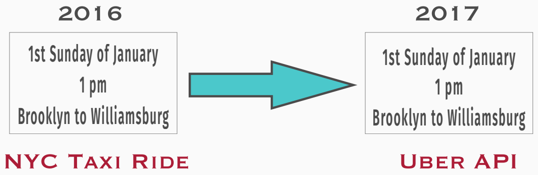
- NYC taxi rides dataset (2015-2016)
  - Over 200,000 taxi rides per day, 29 city-zones.
  - Pickup/ dropoff times and location co-ordinates.

  Used to generate time-evolving empirical transition matrix (F)

- Uber Rides API
  - Provides estimates for price and travel time.
  - Surge multiplier.

  Used to generate rewards matrix (R) and travel time matrix (T)

- Methodology:



Data Processing

- Generation of input matrices:
  - Discretize time into 10 minute steps i.e. $N = 144$ for a day long strategy.
  - For time-evolving matrices $F^{(t)}$, $R^{(t)}$ and $T^{(t)}$, aggregate observations within a 30-minute sliding window centered around time step t.
  - Use passenger arrival and taxi arrival rates to compute probability of finding a passenger at a particular time.

Data Processing

- **Generation of input matrices:**
  - Discretize time into 10 minute steps i.e. $N = 144$ for a day long strategy.
  - For time-evolving matrices $F^{(t)}$, $R^{(t)}$ and $T^{(t)}$, aggregate observations within a 30-minute sliding window centered around time step t.
  - Use passenger arrival and taxi arrival rates to compute probability of finding a passenger at a particular time.
- **Concerns:**
  - Does NYC taxi data reflect same trends as Uber?
  - Endogenous relationship of surge.

Data Processing

- **Generation of input matrices:**
  - Discretize time into 10 minute steps i.e. $N = 144$ for a day long strategy.
  - For time-evolving matrices $F^{(t)}$, $R^{(t)}$ and $T^{(t)}$, aggregate observations within a 30-minute sliding window centered around time step t.
  - Use passenger arrival and taxi arrival rates to compute probability of finding a passenger at a particular time.
- **Concerns:**
  - Does NYC taxi data reflect same trends as Uber? L. Chen, Mislove, and Wilson (2015) have explored this!
  - Endogenous relationship of surge.

Data Processing

- Generation of input matrices:
  - Discretize time into 10 minute steps i.e. $N = 144$ for a day long strategy.
  - For time-evolving matrices $F^{(t)}$, $R^{(t)}$ and $T^{(t)}$, aggregate observations within a 30-minute sliding window centered around time step t.
  - Use passenger arrival and taxi arrival rates to compute probability of finding a passenger at a particular time.
- Concerns:
  - Does NYC taxi data reflect same trends as Uber? L. Chen, Mislove, and Wilson (2015) have explored this!
  - Endogenous relationship of surge.

Better data can be easily incorporated!

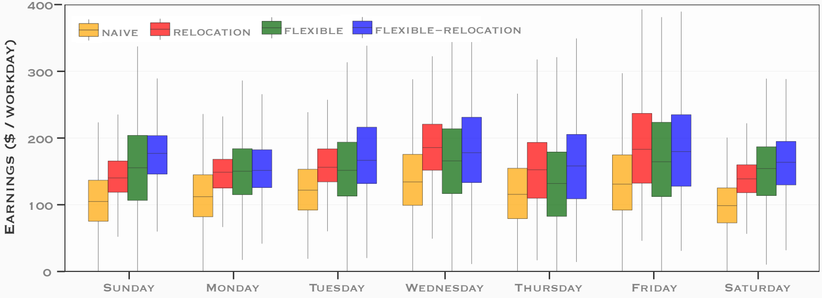
Performance of Strategies

- 100 simulated drivers for each strategy.
  - Random home zone.
  - 8 hour work-day i.e. $B = 48$.
  - Non-flexible strategy drivers work from 9AM-5PM.
  - Train policy on data from week_i.
  - Simulate driver with above policy in week_{i+1}.

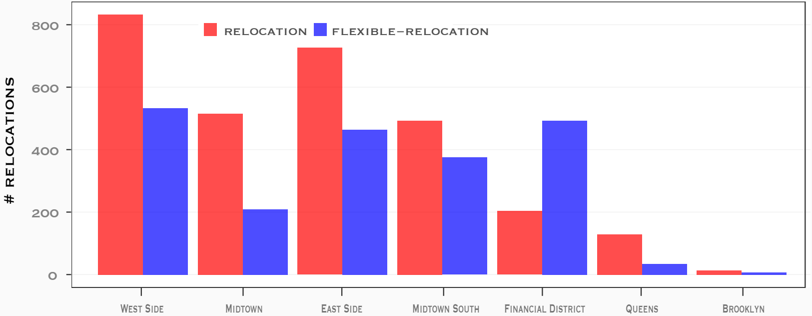
Performance of Strategies

- 100 simulated drivers for each strategy.
  - Random home zone.
  - 8 hour work-day i.e. $B = 48$.
  - Non-flexible strategy drivers work from 9AM-5PM.
  - Train policy on data from week_i.
  - Simulate driver with above policy in week_{i+1}.

Average 47% increase in median earnings per day!

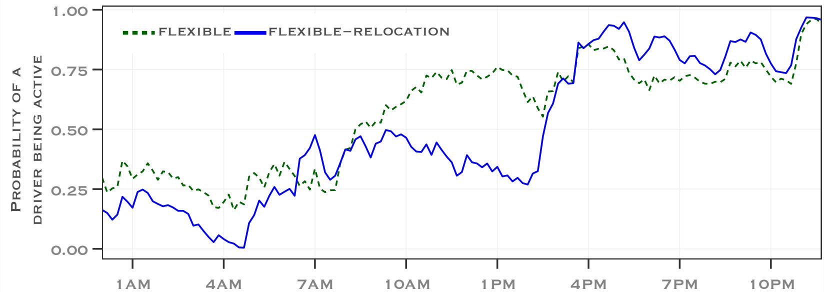


Best zones to re-locate to



Relocation strategy drivers: exploit spatial variation in Manhattan.

Best time to drive



Flexible-Relocation strategy drivers: exploit the temporal variation in Manhattan.

To chase surge or not?

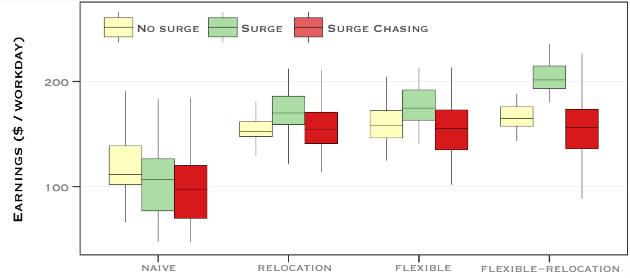
- Anecdotally, Uber drivers 'chase surge'.

To chase surge or not?

- Anecdotally, Uber drivers ‘chase surge’.
- **Simulate surge chasing:** Re-locate to a surging zone if within a 10 minute ride distance

To chase surge or not?

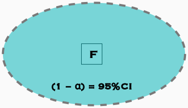
- Anecdotally, Uber drivers ‘chase surge’.
- **Simulate surge chasing:** Re-locate to a surging zone if within a 10 minute ride distance



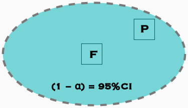
Surge Chasing may lead to lower earnings!

Robustness of Strategies

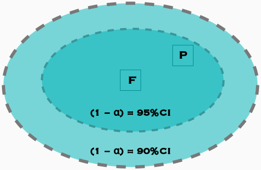
Robust Earnings Problem



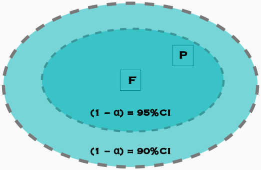
Robust Earnings Problem



Robust Earnings Problem

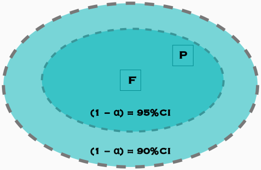


Robust Earnings Problem



- $\mathcal{P}_\alpha$: set of transition matrices within $(1-\alpha)$ CI of observed F i.e. α uncertainty.

Robust Earnings Problem

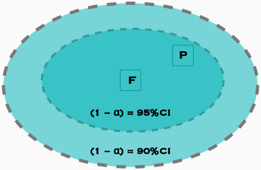


- $\mathcal{P}_\alpha$: set of transition matrices within $(1-\alpha)$ CI of observed F i.e. α uncertainty.

ROBUSTEARNINGS PROBLEM: Find a policy π^* such that:

$$\pi^* = \arg\max_{\pi \in \Pi} \min_{P \in \mathcal{P}_\alpha} \mathcal{E}(\pi, P, T, R, B).$$

Robust Earnings Problem



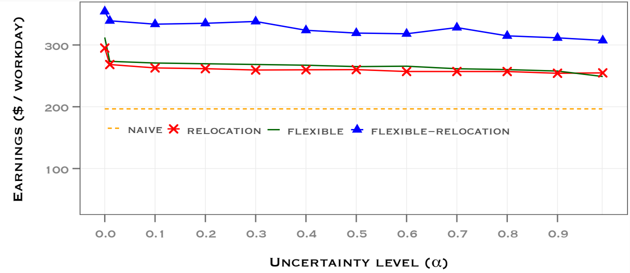
- $\mathcal{P}_\alpha$: set of transition matrices within $(1-\alpha)$ CI of observed F i.e. α uncertainty.

RobustEarnings Problem: Find a policy π^* such that:

$$\pi^* = \arg\max_{\pi \in \Pi} \min_{P \in \mathcal{P}_\alpha} \mathcal{E}(\pi, P, T, R, B).$$

Nilim and El Ghaoui (2004): DP augmented with an optimization step, solvable in polynomial time.

Effect of Uncertainty



Flexible-Relocation Strategy outperforms all other strategies even in presence of high uncertainty!

Conclusion

- Ride-hailing platform's matching algorithms may not be optimal for drivers.
- We propose a deployable methodology for maximizing driver earnings.

Conclusion

- Ride-hailing platform's matching algorithms may not be optimal for drivers.
- We propose a deployable methodology for maximizing driver earnings.

- Need for co-ordination when a significant percentage of driver population becomes strategic.

Thank You!

BOSTON UNIVERSITY

References

Kostas Bimpikis, Ozan Candogan, and Saban Daniela. ‘Spatial pricing in ride-sharing networks’. In: (2016).

Siddhartha Banerjee, Ramesh Johari, and Carlos Riquelme. ‘Pricing in Ride-Sharing Platforms: A Queueing-Theoretic Approach’. In: Abstract appeared in ACM EC-2015. 2015.

Colin Camerer et al. ‘Labor Supply of New York City Cabdrivers: One day at a Time’. In: *The Quarterly Journal of Economics* 112.2 (1997), pp. 407–441.

Juan Camilo Castillo, Dan Knoepfle, and Glen Weyl. 'Surge pricing solves the wild goose chase'. In: *Proceedings of the 2017 ACM Conference on Economics and Computation*. ACM. 2017, pp. 241–242.

Le Chen, Alan Mislove, and Christo Wilson. 'Peeking beneath the hood of Uber'. In: *Proceedings of the 2015 ACM Internet Measurement Conference*. ACM. 2015, pp. 495–508.

M Keith Chen and Michael Sheldon. 'Dynamic Pricing in a Labor Market: Surge Pricing and Flexible Work on the Uber Platform'. In: Working paper. Abstract appeared in ACM EC-2016. 2016.

References iii

Jonathan V. Hall and Alan B. Krueger. *An Analysis of the Labor Market for Uber's Driver-Partners in the United States*. Tech. rep. No. w22843. National Bureau of Economic Research, 2016.

Hongyao Ma, Fei Fang, and David C Parkes. 'Spatio-Temporal Pricing for Ridesharing Platforms'. In: *arXiv preprint arXiv:1801.04015* (2018).

Arnab Nilim and Laurent El Ghaoui. 'Robustness in Markov decision problems with uncertain transition matrices'. In: *Advances in Neural Information Processing Systems*. 2004, pp. 839–846.

Lingyu Zhang et al. ‘A taxi order dispatch model based on combinatorial optimization’. In: *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM. 2017, pp. 2151–2159.