

# An Optimization Framework For Online Ride-sharing Markets

Yongzheng Jia<sup>1</sup>, Wei Xu<sup>1</sup>, and Xue Liu<sup>2</sup>

<sup>1</sup>Institute of Interdisciplinary Information Sciences, Tsinghua University
<sup>2</sup>School of Computer Science, McGill University

 $37^{th}$  ICDCS – June 7, 2017

### ntroduction

Motivation I: Ride-sharing

### Introduction

Problem Mode
Definitions
Task Maps

Offline Alg

Gready Analysis

Online Alg

Max Margir

Simulation

Configurations Results

Conclusion



Emergence and rapid development of online ride-sharing services

## Taxi









# Delivery





Google Express

### Introduction

Motivation II: Two-sided Market

### Introduction

Problem Mode Definitions Task Maps

Offline Alg

Gready Analysis

Online Alg

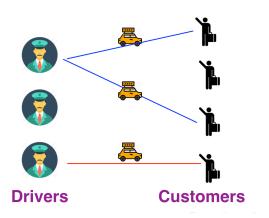
Max Margir

Simulation

Configurations Results



- Two-sided market configuration ⇒ Drivers and Customers
- Existing algorithms are mostly offline heuristics to apply in one-sided market



### Introduction

Motivation III: Inefficiency

### Introduction

Problem Mode
Definitions
Task Maps
Formulations

Offline Alg

Gready Analysis

Online Alg
Nearest
Max Margir

Simulations

Configurations Results

Conclusion



Efficiency of the services are limited by the sub-optimal and imbalanced matching

- Imbalance between supply and demand (e.g. No match or congestion)
- Long waiting time  $\Rightarrow$  Real-time response
- High cost ⇒ Surge Pricing







# Introduction Challenges

### Introduction

Problem Model
Definitions
Task Maps
Formulations

Offline Alg

Gready Analysis

Nearest

Simulations

Configurations Results



- Scalability: Deal with a large number of workers and customers, can partition the map in city's scale (i.e. travel across the entire city)
- Real-time: Always need the platform to give real-time responses to the customers ⇒ Making online algorithms essential

# Introduction Contributions

### Introduction

Problem Model
Definitions
Task Maps
Formulations

Offline Alg

Analysis

Online Al

Nearest

Simulations

Configuration: Results



- Generalized economic models for both Internet taxi and product delivery markets
- A deterministic approximation algorithm with a tight theoretical bound
- Two heuristic online algorithms
- Verify the algorithms with theoretical analysis and trace-driven simulations

Introduction

Problem Model
Definitions

Formulation

Offline Alg

Gready Analysis

Online Alg Nearest

Simulations
Configurations
Results

Conclusion



Two-sided Market with both Temporal and Spatial information

- Drivers The users who provide taxi or delivery services
- Customers The users who receive the services
- Tasks The taxi and delivery services ordered by the customers
- Task Maps DAGs to demonstrate the relationship between the drivers and tasks in the market

Introduction

Problem Mode

### Definitions

Formulations

O(():-- Al-

### Offiline Ai

MDP

. .

Analysis

Online Al

Manuach

Max Margi

#### imulations

Configurations

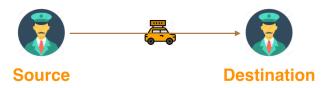
Results

Conclusion



# of drivers: N, and for each driver  $n \in [N]$ :

- lacksquare Source location:  $s_n=(u_n^-,v_n^-)$ , time:  $t_n^-$
- Destination location:  $d_n = (u_n^+, v_n^+)$ , time:  $t_n^+$



Customers and Tasks

ntroduction

Problem Mode

### Definition:

Formulation

Offline Ale

Offiline Alg

MDP

....

Analysis

Online Al

Offilite A

Nearest

Max Margi

imulation

Configuration

Results

Conclusion



# of tasks: M, and for each task  $m \in [M]$ :

- $\blacksquare$  Source location:  $\bar{s}_m = (\bar{u}_m^-, \bar{v}_m^-)$  , time:  $\bar{t}_m^-$
- Destination location:  $\bar{d}_m = (\bar{u}_m^+, \bar{v}_m^+)$ , time:  $t_n^+$
- Price  $p_m$  (calculated by the platform)
- Publishing time  $\bar{t}_m$ :  $\bar{t}_m < \bar{t}_m^- < \bar{t}_m^+$



Source

**Destination** 

Introduction

Problem Mode

Task Maps

Formulation

Offline Alg

Gready

Online Alg

Max Marg

Simulation

Configurations Results

Conclusion



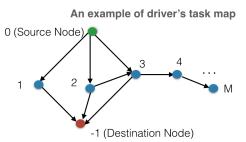


Figure: shows a simple example task map of driver n. The driver can take one task among task 1, task 2 and task 3. She can also take two tasks, and that is to take task 3 after finishing task 2.

Indicator  $h_{n,m,m'} \in \{0,1\}, \forall n \in [N], m,m' \in [\hat{M}]$  denotes whether there is an arc from m to m' in driver n's task map.

Task Map Construction

Introduction

Problem Model
Definitions
Task Maps

Formulation

Offline Alg

Gready

Analysis

Nearest
Max Margi

Simulations

Configuration

Results

Conclusion



 $lacksquare l_{n,m}$ ,  $\hat{c}_{n,m}$  - travel time/cost of the same task m for driver n

- lacksquare  $l_{n,m,m'}$ ,  $c_{n,m,m'}$  travel time/cost of driving empty from m to m' for driver n
- $\hat{h}_{n,m}$  whether driver n can take task m, with  $\hat{h}_{n,m}=1$  indicating a "yes" as follows:

$$\hat{h}_{n,m} = 1 \Leftrightarrow (\hat{l}_{n,m} \le \bar{t}_m^+ - t_m^-), \forall n \in [N], m \in [M]. \tag{1}$$



Task Map Construction

Introduction

Problem Model

Task Maps

Formulations

Offline Alg

MDP

IVIDI

Analysis

Offline Aig

May Marg

Simulations

Configurations

Results

Conclusion



For the arcs from the source (labeled 0) to any task m,

$$h_{n,0,m} = 1 \Leftrightarrow \hat{h}_{n,m} \wedge (l_{n,0,m} \leq \bar{t}_m - t_n^-)$$

$$\wedge (l_{n,m,-1} \leq t_n^+ - \bar{t}_m^+), \quad \forall n \in [N], m \in [M].$$
(2)

Task Map Construction

Introduction

Problem Model

Task Maps

Formulations

Offline Alg

Offine Aig

MDF

Analysis

, mary one

. .

May Margi

simulation

Configuration

Conclusion

( ) 消華大学 Tsinghua University For the arc from one node of task m to the next task m', driver n should have enough time to travel from the destination of task m to the source of task m':

$$h_{n,m,m'} = 1 \Leftrightarrow \hat{h}_{n,m} \wedge \hat{h}_{n,m'} \wedge (l_{n,m',-1} \leq t_n^+ - \bar{t}_{m'}^+) \\ \wedge (l_{n,m,m'} \leq \bar{t}_{m'}^- - \bar{t}_m^+), \forall n \in [N], m \in [M], m' \in [M].$$
(3)

Task Map Construction

Introduction

Problem Model

Task Maps

Formulation

Offline Alg

Gready

Analysis

Online /

Max Margi

imulation

Configuration

Results

Conclusion



If  $h_{n,m,m'}=1$  then also set  $h_{n,m',-1}=1$ , there is an arc from m to m' and another arc from m' to -1.

It will take  $(M^2+2M)$  iterations to calculate all the values of  $h_{n,m,m'}$  for driver  $n\Rightarrow$  Complexity to construct the task map of all the N drivers is  $O(NM^2)$ .

Drivers' Profits Maximization: Objective

Introductio

Problem Mode Definitions

Formulation

Offline Alg

MDP

Analysis

Online Alg

Max Margir

Simulation

Configuration

Conclusion

简单大学 Tsinghua University Our Goal: Maximize drivers' total profits  $\Rightarrow$  Total Revenue - Total Excess Cost (Shown in (4))

### Decision variables:

- $x_{n,m}$  If task m is assigned to driver n in the market
- $y_{n,m,m'}$  If driver n takes task m' after finishing task m.

$$Z: \text{maximize} \sum_{n \in [N]} \sum_{m \in [M]} x_{n,m} p_m - \Big(\sum_{n \in [N]} \sum_{m \in [M]} x_{n,m} \hat{c}_{n,m} + \Big(\sum_{m \in [M]} \sum_{m \in [M]} x_{m,m} \hat{c}_{m,m} + \Big(\sum_{m \in [M]} \sum_{m \in [M]} x_{m,m} \hat{c}_{m,m} + \Big(\sum_{m \in [M]} \sum_{m \in [M]} x_{m,m} \hat{c}_{m,m} + \Big(\sum_{m \in [M]} \sum_{m \in [M]} x_{m,m} \hat{c}_{m,m} + \Big(\sum_{m \in [M]} \sum_{m \in [M]} x_{m,m} \hat{c}_{m,m} + \Big(\sum_{m \in [M]} \sum_{m \in [M]} x_{m,m} \hat{c}_{m,m} + \Big(\sum_{m \in [M]} \sum_{m \in [M]} x_{m,m} \hat{c}_{m,m} + \Big(\sum_{m \in [M]} \sum_{m \in [M]} x_{m,m} \hat{c}_{m,m} + \Big(\sum_{m \in [M]} \sum_{m \in [M]} x_{m,m} \hat{c}_{m,m} + \Big(\sum_{m \in [M]} \sum_{m \in [M]} x_{m,m} \hat{c}_{m,m} + \Big(\sum_{m \in [M]} \sum_{m \in [M]} x_{m,m} \hat{c}_{m,m} + \Big(\sum_{m \in [M]} \sum_{m \in [M]} x_{m,m} \hat{c}_{m,m} + \Big(\sum_{m \in [M]} \sum_{m \in [M]} x_{m,m} \hat{c}_{m,m} + \Big(\sum_{m \in [M]} \sum_{m \in [M]} x_{m,m} \hat{c}_{m,m} + \Big(\sum_{m \in [M]} x_{m,m} + \Big(\sum_{m \in [M]} x_{m$$

$$+ \sum_{n \in [N]} \sum_{m \in [\hat{M}]} \sum_{m' \in [\hat{M}]} y_{n,m,m'} h_{n,m,m'} c_{n,m,m'} - \sum_{n \in [N]} c_{n,0,-1} \Big).$$

(4)

Drivers' Profits Maximization: Constraints

s.t.

introduction

Problem Model

Definitions

Formulation

Offline Alg

Croady

Analysis

Online Alg

Max Margin

Configurations

Results

Conclusion

( ) 消華大学 Tsinghua University

$$\sum_{n \in [N]} x_{n,m} \le 1, \quad \forall m \in [M]; \tag{5a}$$

$$\sum_{m \in [M]} x_{n,m} p_m \ge \sum_{m \in [\hat{M}]} \sum_{m' \in [\hat{M}]} y_{n,m,m'} h_{n,m,m'} c_{n,m,m'}$$

+ 
$$\sum x_{n,m} - c_{n,0,-1}, \forall n \in [N];$$

$$+\sum_{m\in[M]}x_{n,m}-c_{n,0,-1}, \forall n\in[N],$$

$$\sum_{n' \in [\hat{M}]} h_{n,0,m'} y_{n,0,m'} = 1, \quad \forall n \in [N];$$

$$\sum_{m \in [\hat{M}]} h_{n,m,-1} y_{n,m,-1} = 1, \quad \forall n \in [N];$$
 (5)

(5a): task allocation, (5b): individual rationality (5c)-(5d): flow conservation for sources and destinations

(5b)

## Problem Model I:

Drivers' Profits Maximization: Constraints (Cont'd

ntroduction

Problem Model

Deminitions

Formulations

Offline Alg

MDP

O-1'-- Al-

Mearest

Max Margir

Simulations

Configuration

Conclusion



$$\sum_{m \in [\hat{M}]} h_{n,m,m'} y_{n,m,m'} = x_{n,m'}, \forall n \in [N], m' \in [M];$$
 (6a)

$$\sum_{(n,m)\in I} h_{n,m,m'} y_{n,m,m'} = x_{n,m}, \forall n \in [N], m \in [M];$$
 (6b)

$$x_{n,m} \in \{0,1\}, \quad \forall n \in [N], m \in [M];$$
 (6c)

$$y_{n,m,m'} \in \{0,1\}, \quad \forall n \in [N], m \in [\hat{M}], m' \in [\hat{M}].$$
 (6d)

(6a)-(6b): flow conservation for internal nodes (6c) - (6d): decision variables

# Problem Model II:

Social Welfare Maximization

ntroduction

Problem Model

Task Maps

Formulations

Offline A

MDP

Gready

Analysis

Online /

N.4 --- N.4 ----

Simulation

Configuration

Results

Conclusion



 $b_m$ : Customers' willingness-to-pay for task m

$$\hat{Z} : \text{maximize} \sum_{n \in [N]} \sum_{m \in [M]} x_{n,m} b_m - \Big(\sum_{n \in [N]} \sum_{m \in [M]} x_{n,m} \hat{c}_{n,m}$$

$$+ \sum_{n \in [N]} \sum_{m \in [\hat{M}]} \sum_{m' \in [\hat{M}]} y_{n,m,m'} h_{n,m,m'} c_{n,m,m'} - \sum_{n \in [N]} c_{n,0,-1} \Big).$$

s.t

Previous Constrains +

$$\sum_{n \in [N]} x_{n,m} (b_m - p_m) \ge 0, \forall m \in [M]. \tag{8}$$

# Problem Models Solving Ideas

#### Introduction

Problem Mode

Task Maps

Formulation

Offline Alg

Gready

Analysis

Unline Alg

Max Margi

Simulation

Configurations Results



- Solving (4) or (7) is NP-hard
- In the real markets, it is hard to formulate the social welfare, since it is hard to estimate  $b_m$
- Optimizing the drivers' total profits is enough to improve the efficiency of the ride-sharing markets
- Relax to LP and get an upper bound of OPT

# Offline Approximation Algorithm

Introduction

Problem Model
Definitions
Task Maps
Formulations

Offline

### MDP

Analysis

Online Alg

Max Margin

Simulation

Configuration:



- Original Problem: Allocate tasks to drivers for total profits maximization (temporal + spatial)
- Merge all the N task maps into one DAG (G). Assign each task to at most one driver (Node-disjoint needed).
- Objective: Find multiple weighted node-disjoint paths with maximum total value.
- EDP: Edge-disjoint paths (existing solutions)

# Offline Approximation Algorithm

Introduction

Problem Mode

Task Maps

Formulation

Offline A

MDP

Analysis

Online Alg Nearest

Simulation

Configuration

Conclusion



### Definitions:

- $\blacksquare$   $\pi$ : A path from a source to a destination
- lacksquare  $\mathcal{P}_i$ : All the paths in the graph G from  $s_i$  to  $d_i$  for driver i
- $f_{\pi}$ :Whether path  $\pi$  is selected in the solution
- $r_{\pi}$ : Profit of the path the summation of the total value of the tasks subtracting the excess cost (defined in Eq. (4))

$$Z: \mathsf{maximize} \sum_{\pi \in \cup_i \mathcal{P}_i} f_{\pi} r_{\pi}. \tag{9}$$

s.t.

$$\sum_{\pi \in \mathcal{P}_i} f_{\pi} = x_i, \forall i \in [N]; \tag{10a}$$

$$\sum_{i=1} \sum_{\pi \in \mathcal{P}_i : m \in \pi} f_{\pi} \le 1, \forall m \in [M];$$

$$x_i \in \{0, 1\} \forall i \in [N];$$

$$f_{\pi} \in \{0,1\}, \forall \pi \in \cup_i \mathcal{P}_i.$$

(9): Same as (4), for the drivers' total profits

(10a): Each driver may choose 1 or 0 task list (10b): Node-disjoint guarantee

(10b)

(10c)

(10d)

# Offline Approximation Algorithm The Greedy Algorithm: Pseudocode

Introduction

Problem Mo Definitions

Formulations

Offline Al

MDP

Analysis

Online Alg

Max Margin

Configuration

Results

Conclusion



Initialization: Let  $S=\emptyset$ ,  $\Pi=\emptyset$ ,  $X=\{1,2,\cdots,N\}$ , G'=G while there exists driver  $i\in X$  and path  $\pi\in \cup_i \mathcal{P}_i$  from  $s_i$  to  $d_i$  with strictly positive profit  $r_\pi>0$  do

- (a) Find the path  $\pi^* = argmax_{\pi \in \cup_i \mathcal{P}_i} r_{\pi}$ , such that  $\pi^*$  has the maximum profit in the current graph G'. Let  $\pi^*$  be the task list for driver  $i^*$ :
- (b) Remove the source and destination nodes  $(s_{i^*}, d_{i^*})$  of driver  $i^*$  and all the task nodes in  $\pi^*$  from the current graph G';
- (c)  $S = S \cup i^*$ ,  $\Pi = \Pi \cup \pi^*$ ,  $X = X/i^*$ ;

### end

Output the drivers in set S and the selected paths (i.e. task lists) in  $\Pi$ .

# Offline Approximation Algorithm Theoretical Analysis

Introduction

Problem Model

Task Maps

Formulation

Offline A

MDP

Analysis

Analysi

Online A

Nearest

Max Margi

Simulation

Configuration

Results

Conclusion



### **Theorem**

The Greedy Algorithm (i.e. GA) gives a feasible solution with  $(\frac{1}{D+1})$ -approximation ratio in polynomial time, where D is the maximum number of nodes in a path (i.e. the diameter of the graph G). The ratio is tight.

# Offline Approximation Algorithm

ntroduction

Problem Model

Definitions

Formulations

Offline A

MDP

Gready

Analysis

Online Alg

May Marg

imulation

Configurations

Results

Conclusion



## Lemma 1: Complexity

GA achieves a feasible solution of (4) within time complexity  ${\cal O}(N^2M^2).$ 

# Offline Approximation Algorithm

Introduction

Problem Model

Definitions

Farmulations

Offline A

MDD

Gready

Analysis

Online Alg

Ollillic Alg

Max Margir

Simulations

Configurations

Results

Conclusion



## Lemma 2: Lower Bound

*GA* guarantees an approximation ratio of  $(\frac{1}{D+1})$ .

# Offline Approximation Algorithm Theoretical Analysis

Introduction

Problem Mode

Task Maps

Formulations

MDP

Crondy

Analysis

Online Alg

Max Margir

Simulations

Configuration Results

Conclusion



- B: Set of paths selected by GA
- O: Paths selected by the optimal solution (i.e. OPT)
- GA terminates in K iterations,  $\left\{\pi_k\right\}_{k=1,2,\cdots,K}$  is the path selected by GA during the k-th iteration.

### Proposition 1

Every path in  $\mathcal{O}$  must intersect with at least one path in  $\mathcal{B}$ .

## Proposition 2

Every path in  $\mathcal{B}$  intersects with at most (D+1) paths in  $\mathcal{O}$ .

# Offline Approximation Algorithm

Introduction

Problem Mode

Task Maps

Offline Alg

MDP

Gready

Online Alg

Nearest

iviax iviargin

Configuration

Conclusion



 $\mathcal{O}_k$  : Set of paths in  $\mathcal O$  that intersect with  $\pi_k$ 

Proposition 3

$$\mathcal{O} = \cup_{k=1}^K \mathcal{O}_k$$

Proposition 4

$$\sum_{\pi \in \mathcal{O}} r_{\pi} \le (D+1) \cdot r_{\pi_k}, \quad \forall k = 1, 2, \cdots, K$$
 (12)

(11)

# Offline Approximation Algorithm

### ntroduction

Problem Mode

Task Maps

Formulations

Offline Al

IVIDE

Analysis

### Online Alg

Nearest

IVIAX IVIAIR

Simulations

Configuration

Results

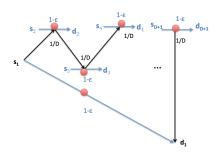
Conclusion



## Lemma 3: Upper Bound

 $(\frac{1}{D+1})$  is also the upper bound to the approximation ratio.

- $\mathcal{O}$  chooses Blue Edges  $\Rightarrow (D+1) \cdot (1-\epsilon)$  (OPT)
- $\mathcal{B}$  chooses Black Edges  $\Rightarrow 1$  (GA)



# Offline Approximation Algorithm Discussions

### Introduction

Problem Model
Definitions

Task Maps Formulations

Offline Alg

Gready

Analysis

Online Alg

Max Margii

Simulation

Configuration: Results



- Motivated by the EDP model, state-of-the-art bound:  $O\Big(min(n^{2/3},\sqrt{m})\Big) \text{ for undirected graphs and } \\ O\Big(min(n^{4/5},\sqrt{m})\Big) \text{ for directed graph.}$
- $(\frac{1}{D+1})$  is a tight bound, and can apply well in real markets. D is small for ride-sharing. D=1 and  $\frac{1}{2}$  approximation ratio for Google's Waze Rider market.

# Online Heuristic Algorithms

Heuristic I: Nearest Drivers

### Introduction

Problem Model

Definition.

Formulation

\_\_\_\_

Offiline

MDP

Gready

Analysis

Online A

Nearest

Max Margii

Simulation

Configuration

Results



- lacktriangle When a task m arrives, chooses the driver who can arrive at the first time
- Update the information of tasks and drivers
- If no driver can take the task, then drop task m

# Online Heuristic Algorithms

Heuristic I: Nearest Drivers

Introduction

Problem Mode

Formulations

Formulation

Offline Alg

Gready

Analysis

Online A

Neares

Max Marg

Simulation

Configuration

resuits

Conclusion



## Define the marginal value:

$$\delta_{n,m} = p_m - (c_{n,m,-1} + \hat{c}_{n,m} + c_{n,m',m} - c_{n,m',-1})$$

- When a task m arrives, chooses the driver n who can serve with the largest  $\delta_{n,m}$
- Update the information of tasks and drivers
- $lue{}$  If no driver can take the task, then drop task m

Experiment Setup

Introduction

Problem Model
Definitions
Task Maps
Formulations

Offline Alg MDP Gready

Gready Analysis

Online Alg
Nearest
Max Margin

imulations

Configuration



- Dataset: ECML/PKDD 15 including a complete year (from 01/07/2013 to 30/06/2014) of the trajectories for all the 442 taxis running in the city of Porto, Portugal
- 1,000,000+ records with detailed information, including the timestamp of starting time and finishing time for each trip, polyline of the trip trajectory, and the driver ID

**Experiment Setup** 

#### ntroduction

Problem Model
Definitions

Formulation

Offline A

Croady

Analysis

Online Alg

Max Margi

Simulation

Configuration

Results

Conclusion



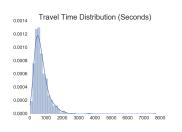


Figure: Travel Time Distribution

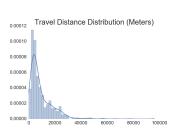


Figure: Travel Distance Distribution

Results: Performance Ratios



Problem Mode

Task Maps

Formulations

Offline Alg

MDP Gready

Analysis

Online Alg

May Mare

Simulation

Configurations

Results

Conclusion



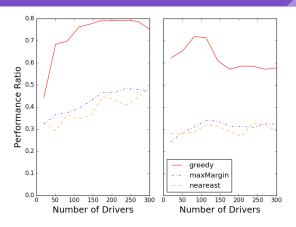


Figure: The left figure shows the performance ratio of the "hitchhiking" model and the right figure shows the performance ratio of the "home-work-home" model

Results: More Insights

Introduction

Problem Mode

Task Maps

Formulatio

Offline /

MDP

Gready

Analysi

Online Alg

Offilite Aig

B.1 . .

Max Margir

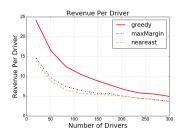
Simulation

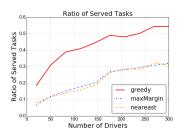
Configurations

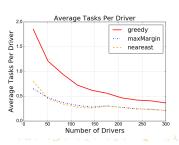
Results











# Thank You

### Introductio

Problem Model
Definitions
Task Maps
Formulations

Offline Alg

Gready Analysis

Nearest
Max Margi

Configurations

Results

Conclusion



### Conclusion Remarks

- Propose generalized economic models for ride-sharing markets: Dynamic Scheduling based on Temporal + Spatial info
- A deterministic offline algorithm + Two online heuristics
- Application Specialization: Limited # of tasks within a period, our greedy algorithm works fine
- Future Work: Design deterministic online algorithms

### Contact Information - Yongzheng Jia

jiayz13@mails.tsinghua.edu.cn

# Thank You



Yongzheng Jia jiayz13@mails.tsinghua.edu.cn Wechat ID: jiayz90