"Where should an empty Uber go?"

— Vehicle rebalancing in Mobility on demand systems

Pengbo Zhu and Nikolas Geroliminis
Urban Transport Systems Laboratory

EPFL | LUTS

Oct. 30 2024

Motivation

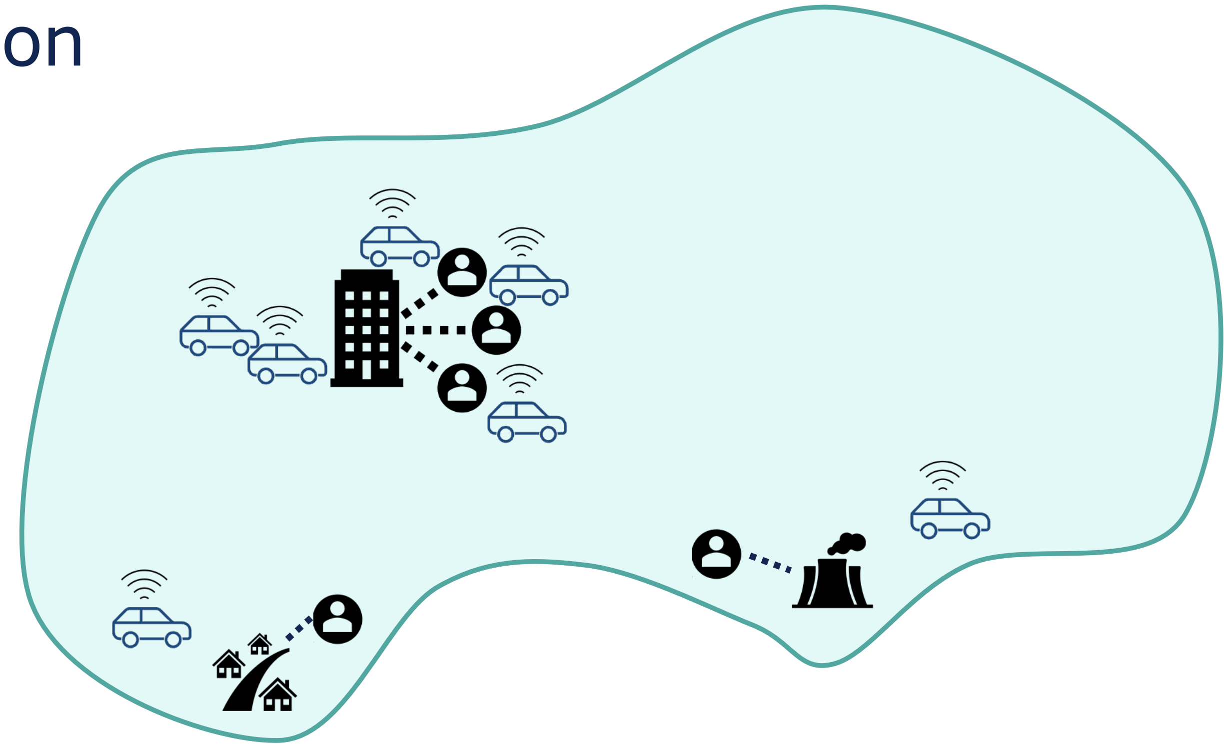
Imbalance between passenger demand and vehicle supply

Goal: Empty vehicle repositioning

- Reposition vehicles to high demand regions

[1] Zuba blog, https://medium.com/zoba-blog/supply-demand-imbalance-in-shared-mobility-cost-and-consequence-72a50007831b

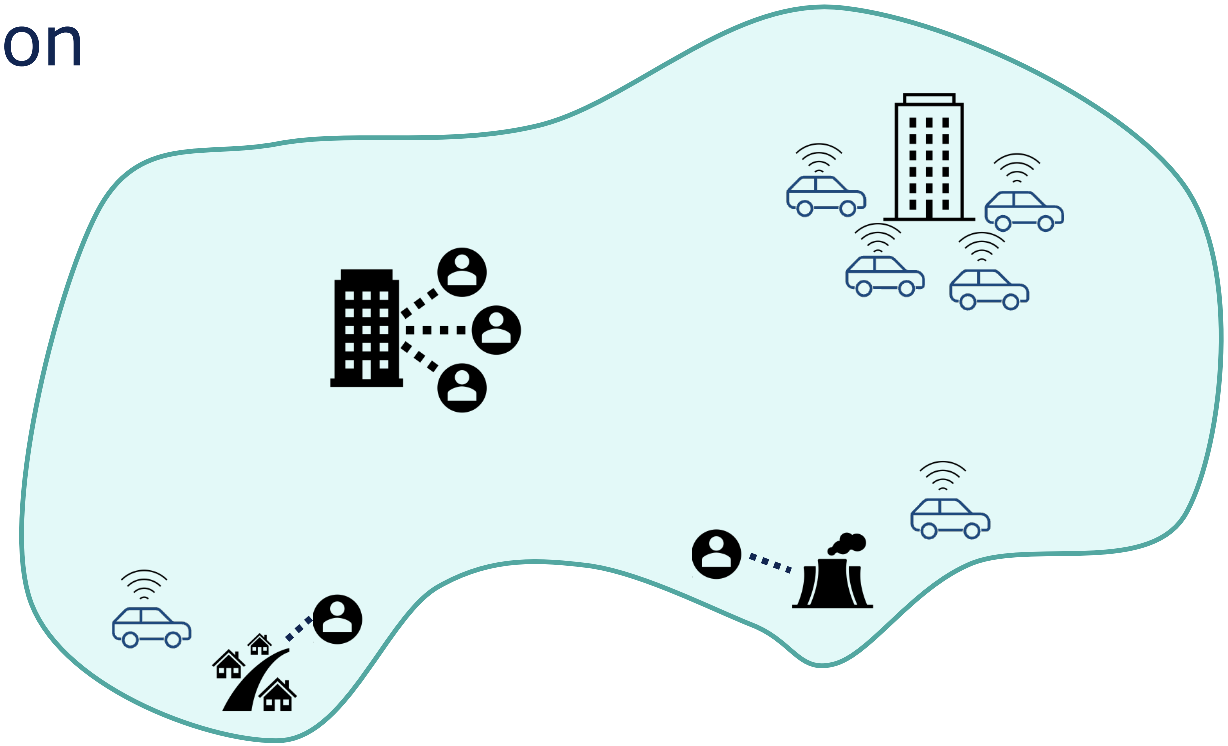
Motivation



Imbalance in the spatial distribution of vehicles:

- Non-uniform passenger's demand for rides in different districts,
- Asymmetry between origin and destination distributions of trips.

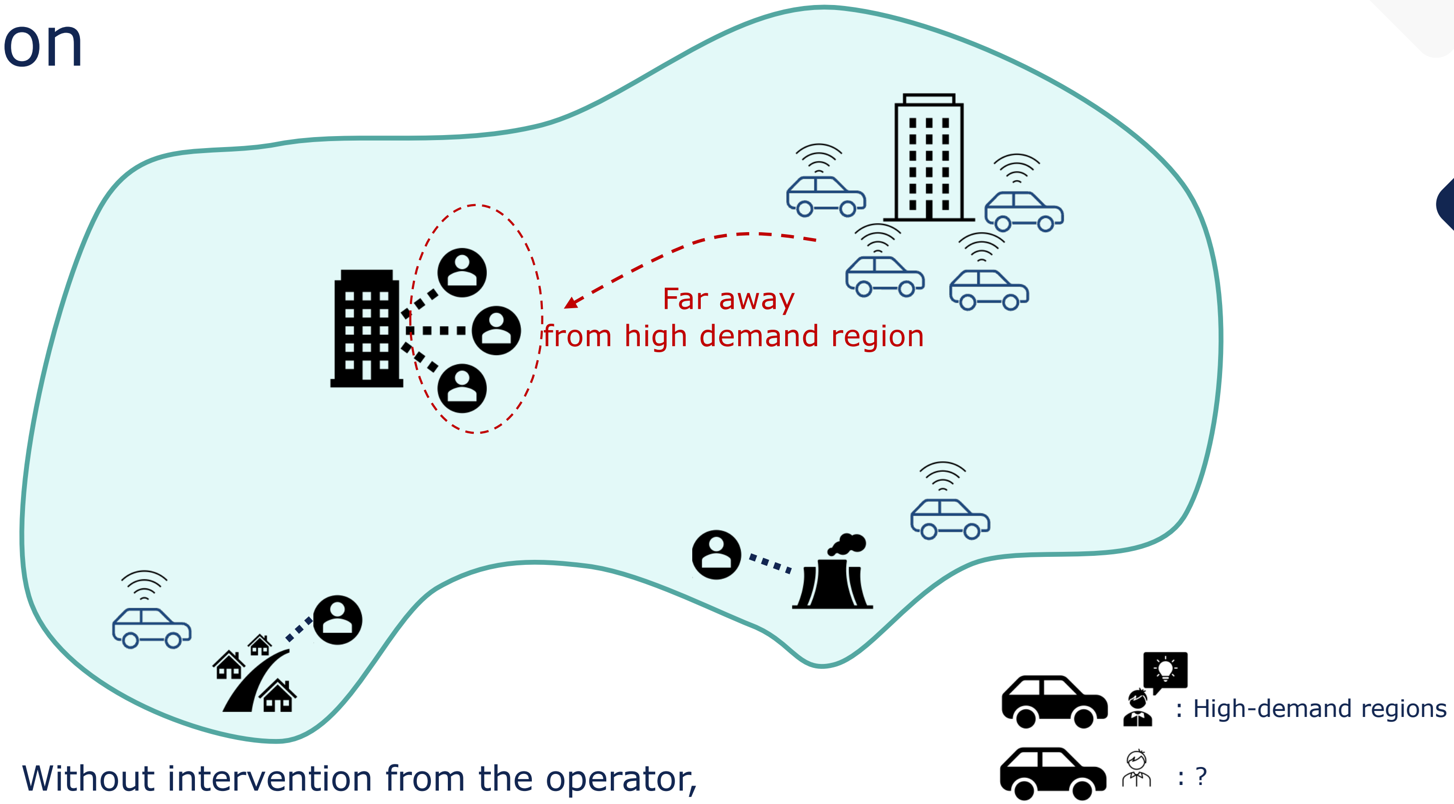
Motivation



Imbalance in the spatial distribution of vehicles:

- Non-uniform passenger's demand for rides in different districts,
- Asymmetry between origin and destination distributions of trips.

Motivation



Without intervention from the operator,

The fleet will be increasingly concentrated in low-demand locations, rather than high-demand locations.

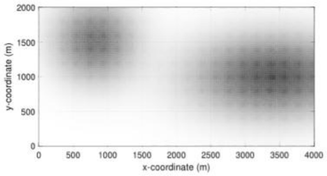
Goal: rebalancing vehicles

- Relocating vehicles to the high-demand regions.

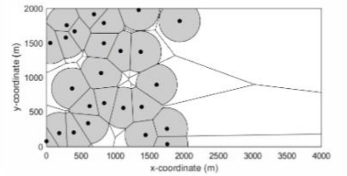
Part I: Idle Vehicle Rebalancing

Idle-vehicle Rebalancing Coverage Control for Ride-sourcing systems[1]

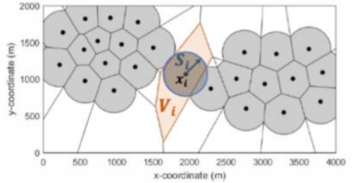
Goal: Align the vehicle distribution with the demand density.



(a) Demand density function



(b) Initial configuration



(c) Final configuration

x_i: position of idle AVs,
n: fleet size of idle AVs,

S_i: covered area,
V_i: Voronoi cell,
$W_i = S_i \cap V_i$.

Vehicle Rebalancing Problem

Coverage Control Problem

Every agent/vehicle x_i is responsible for covering a certain area W_i

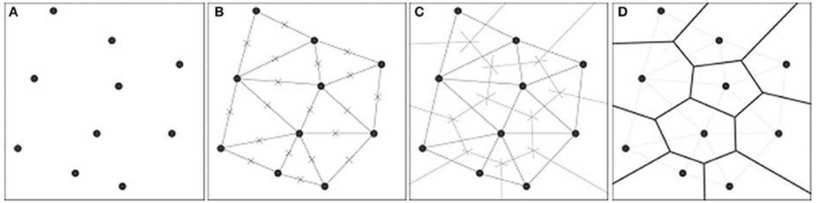
$$H(X,W) = \sum_{i=1}^{n} H(x_i, W_i) = \sum_{i=1}^{n} \int_{q \in W_i} \|x_i - q\|^2 \varphi(q)\, dq$$

$\varphi(q)$: demand density function

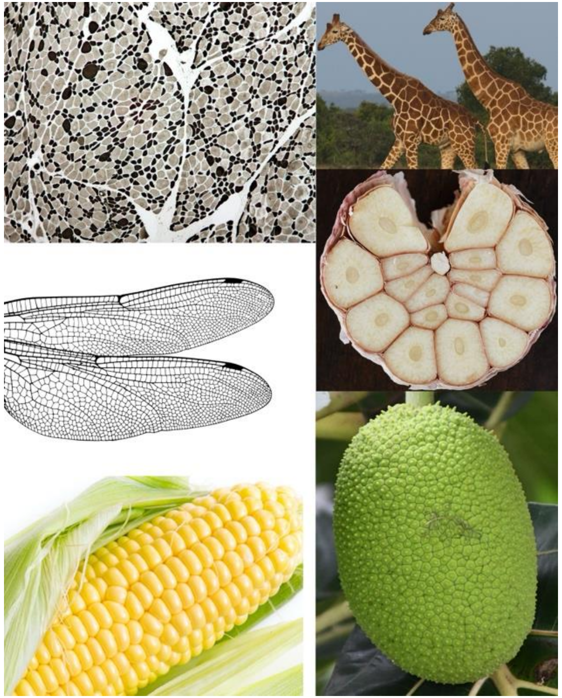
[1] P. Zhu, I.I.Sirmatel, G. Ferrari-Trecate, and N. Geroliminis, "A Coverage Control-based Idle Vehicle Rebalancing Approach for Autonomous Mobility-on-demand Systems," IEEE Transactions on Control Systems Technology, 2024;

Voronoi Partition

- The partitioning of a plane with n points into convex polygons.
- Each cell contains one seed.
- Every point in a given cell is closer to its seed than to any other.



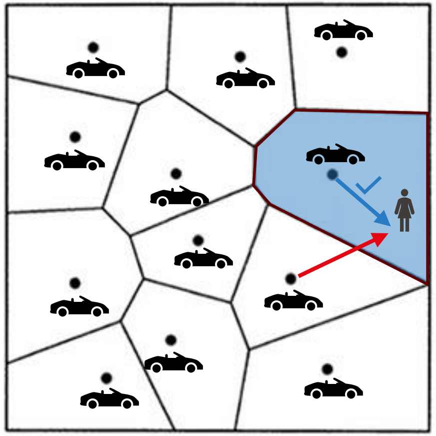
How to construct Voronoi Partition



Voronoi Partition in nature

Voronoi Partition

- The partitioning of a plane with n points into convex polygons.
- Each cell contains one seed.
- Every point in a given cell is closer to its seed than to any other.



Why we use it?
The pick-up task at a point $q \in V_i$ in the Voronoi cell V_i should be executed by the vehicle closest to q, which is exactly the vehicle i.

Distributed Coverage Control Algorithm

Coverage Objective:

$$H(X, W) = \sum_{i=1}^{n} H(x_i, W_i) = \sum_{i=1}^{n} \int_{q \in W_i} \|x_i - q\|^2 \varphi(q) dq$$

where $\varphi(q)$ is the demand density function.

The local minimum of H can be obtained when all x_i are located at centroids $C(W_i)$, i.e., center of mass, of their respective Voronoi cells (W_i), i.e., ***Centroidal Voronoi Configuration(CVC)[1]***.

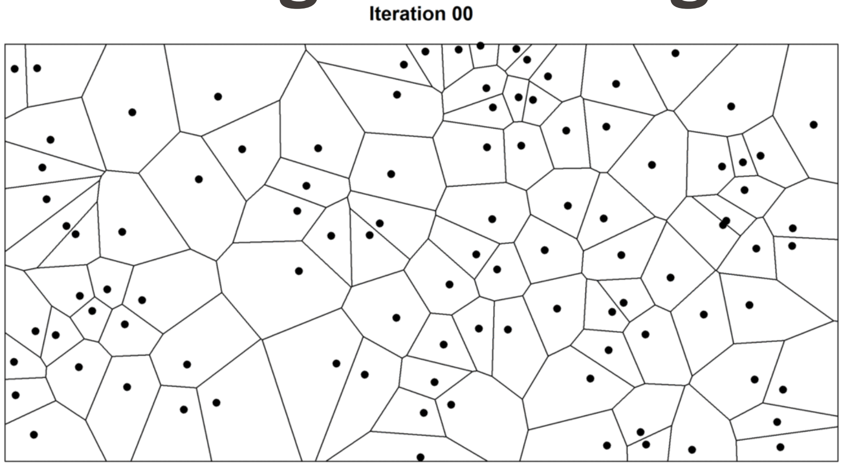
Distributed Control Law:

$$\dot{x}_i = u_i,$$
$$\boldsymbol{u_i} = -\boldsymbol{k_W}(\boldsymbol{x_i} - \boldsymbol{C}(\boldsymbol{W_i})), \qquad \boldsymbol{k_w} > \boldsymbol{0}.$$

Move towards the centroid!

[1] Du et al. (1999), *Centroidal voronoi tessellations: Applications and algorithms.*

Distributed Coverage Control Algorithm



The control law will converge to CVC
Initial position (black dots), current position (circles)

Distributed Control Law:

$$\dot{x}_i = u_i,$$
$$\boldsymbol{u_i} = -\boldsymbol{k_W}(\boldsymbol{x_i} - \boldsymbol{C}(\boldsymbol{W_i})), \qquad \boldsymbol{k_w} > \mathbf{0}.$$

Move towards the centroid!

which ensure ***H is monotonously decreasing***, steers the fleet to ***converge to CVC.***

Pengbo ZHU

[1] Du et al. (1999), *Centroidal voronoi tessellations: Applications and algorithms.*

Distributed Coverage Control Algorithm

Distributed Control Law:

$$\dot{x}_i = u_i,$$
$$\boldsymbol{u_i} = -\boldsymbol{k_W}(\boldsymbol{x_i} - \boldsymbol{C}(\boldsymbol{W_i})), \qquad \boldsymbol{k_w} > \boldsymbol{0}.$$

Move towards the centroid!

It can be operated in a ***distributed*** manner:

If each agent can ***communicate with its neighbors***,
leveraging ***local information***, i.e., coordinates of its neighbors.

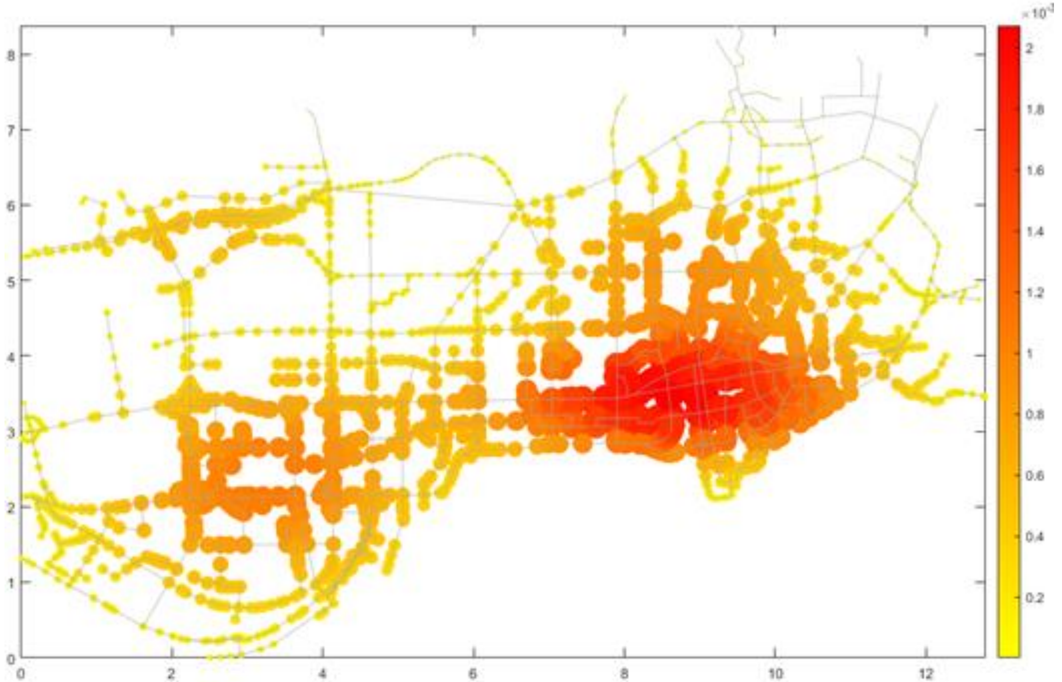
[1] Du et al. (1999), *Centroidal voronoi tessellations: Applications and algorithms.*

Case Study

Simulated network:
- Shenzhen, China
- 1858 nodes
- 2013 links

Beojone, Caio & Geroliminis, Nikolas. (2020). *On the inefficiency of ride-sourcing services towards urban congestion.*

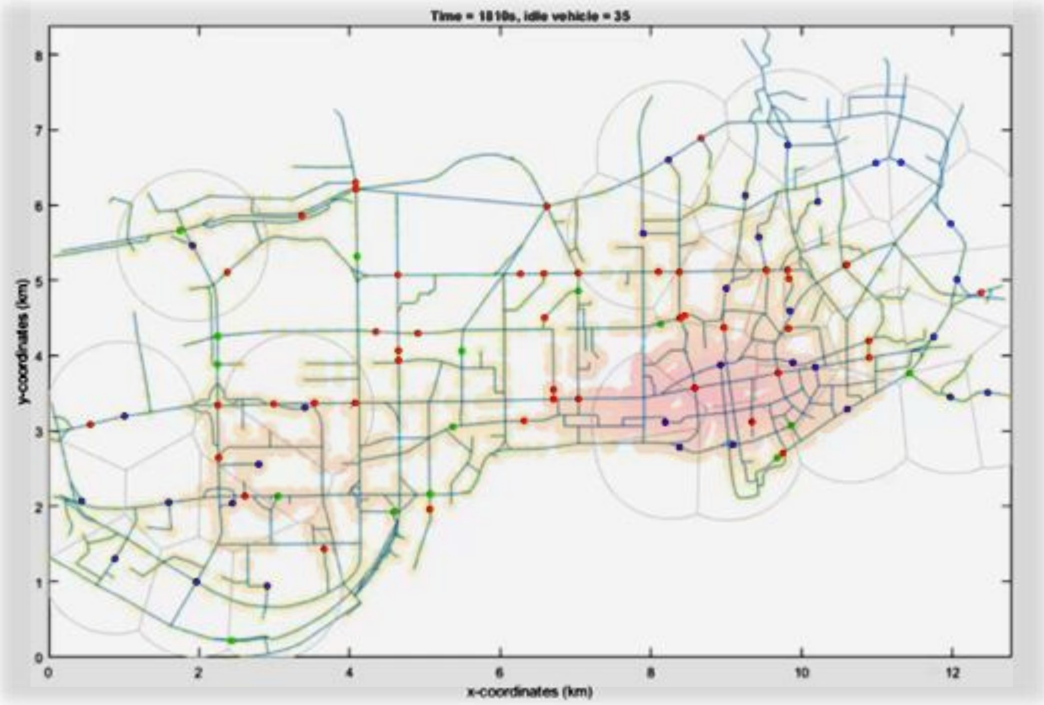
Demo Video



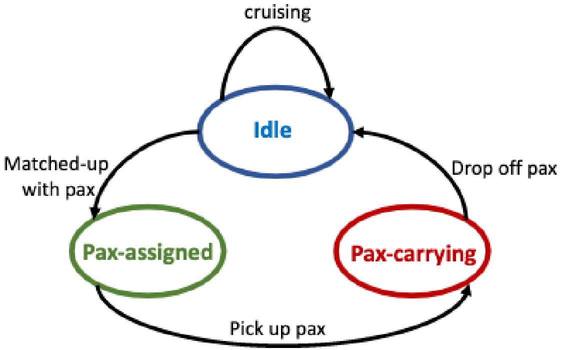
Demand distribution

Pengbo ZHU

Demo Video



blue: ***idle vehicle (i.e., empty, looking for a passenger),***
green: passenger-assigned vehicle,
red: passenger-carrying vehicle.



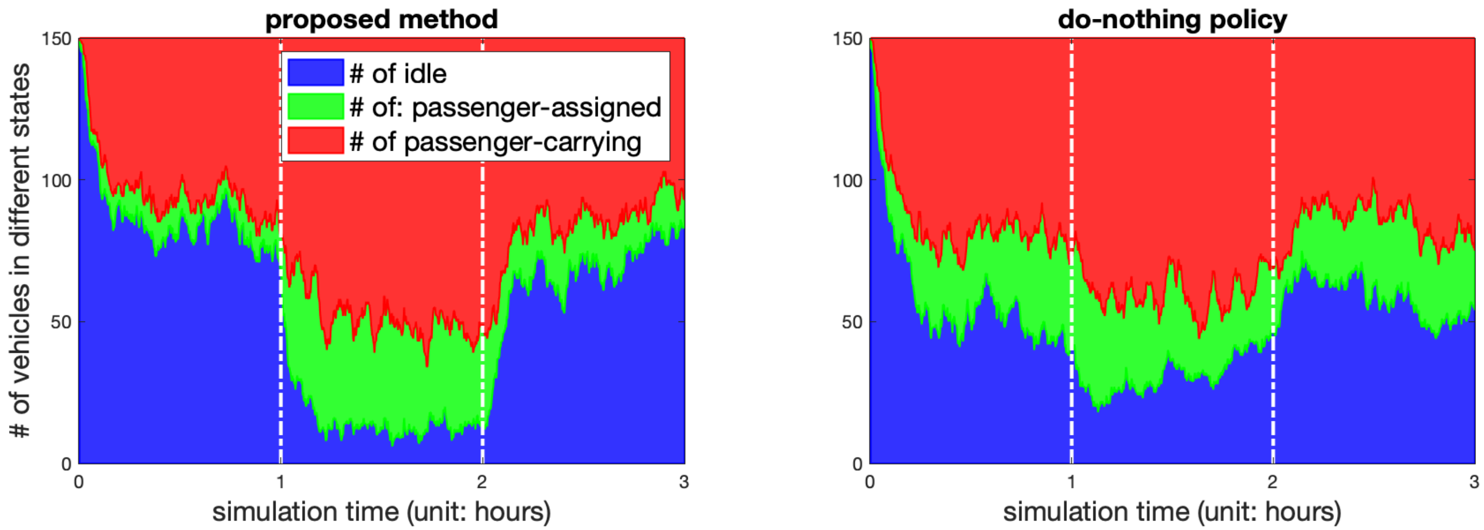
Results

- 3-hour simulation
- Time Pattern of demand: low-high-low, each period lasts for 1 hour
- 2400 orders
- Fleet size = 150

	CVR	**Do-nothing**	**Improvement**
Answer rate (%)	82.9	73.2	13.3%↑
Average waiting time (s)	132.5	173.9	23.8%↓

NB:
1. CVR: **C**overage Control for **V**ehicle **R**ebalancing;
2. Do-nothing: AVs stop at their last destinations, until they are matched with passengers again.

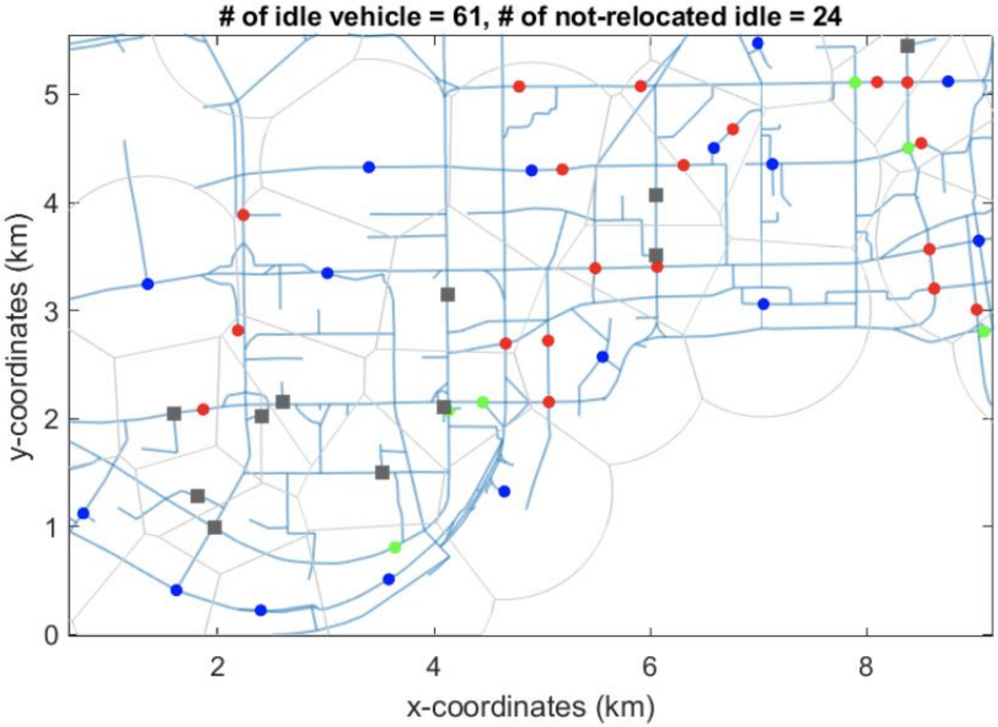
Results



Comparison of different states of vehicles

- Operate the fleet more efficiently as a larger amount of vehicles are actively serving passengers

Extension: Is it necessary to make all empty vehicles follow coverage control?



Blue dot: active idle

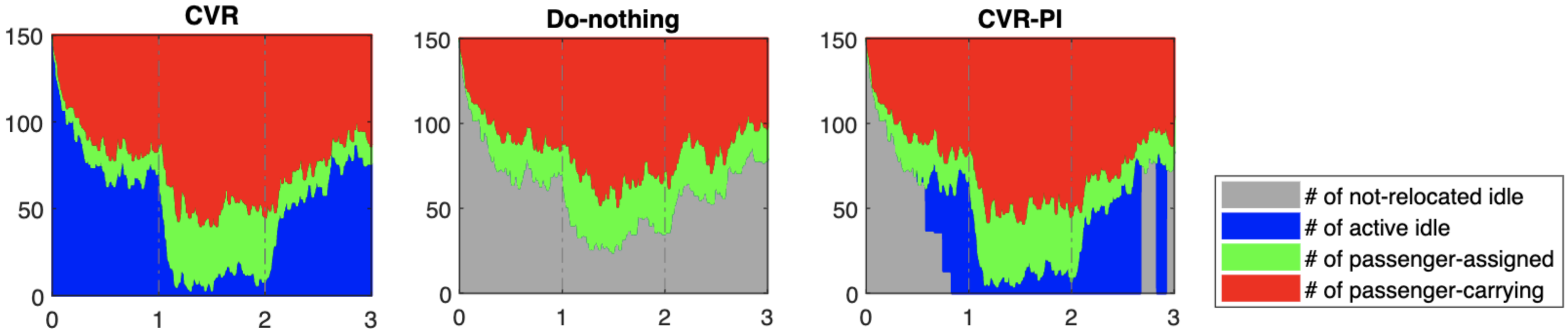
Gray square: Not-relocated idle

Extension: Is it necessary to make all empty vehicles follow coverage control?

Pros
- Enhanced answer rate
- Reduced waiting times

Cons:
- Extra energy or fuel cost
- Congestion
- Emissions

Extension: Is it necessary to make all empty vehicles follow coverage control?



	Answer rate (%)	Average waiting time (s)	Rebalancing distance (km)
CVR	**82.6**	127.0	**4804.8**
CVR-PI	**82.4**	139.4	**2314.7**
Do-nothing	73.1	155.1	0

Summary

Distributed Coverage Control for Vehicle Rebalancing (CVR):

Pros:

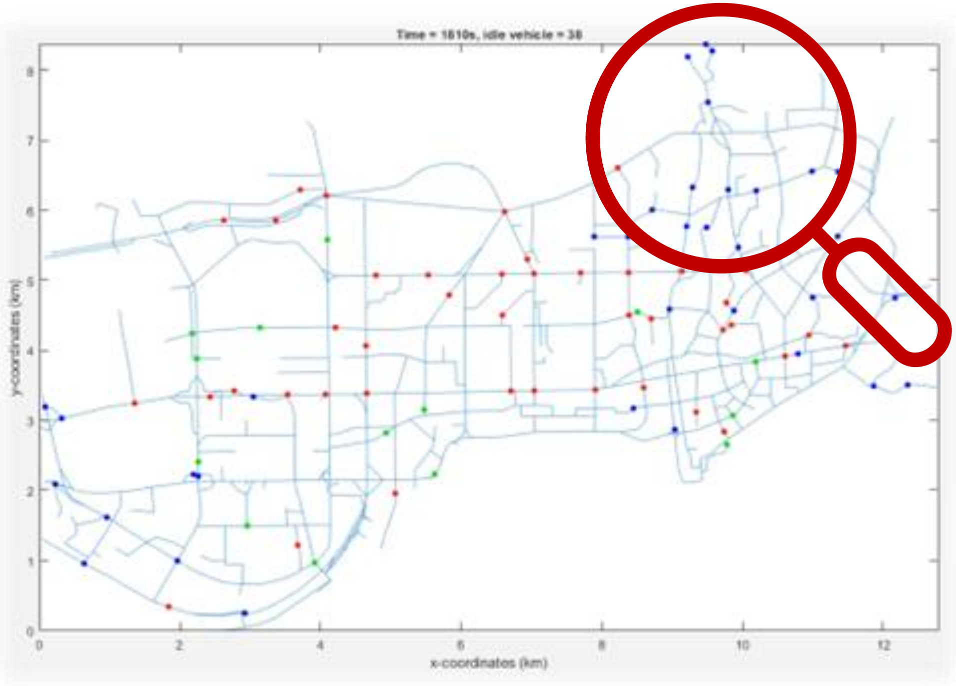
- Node-level route guidance;
- CVR can be implemented by agent itself to gain its own control actions.
- Suitable for large-scale system thanks to the **scalability** and **distributed property**.

Summary

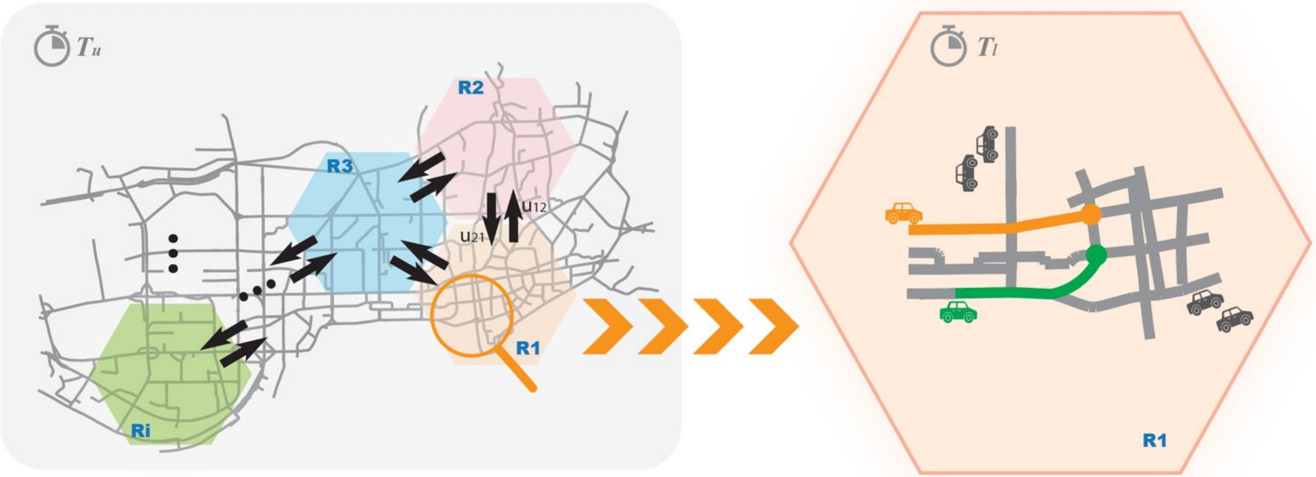
Distributed Coverage Control for Vehicle Rebalancing (CVR):

Cons:

- It lacks coordination of other regions in the network.



Hierarchical Control Framework for Repositioning



High Level (Regional Level)

Low Level (Vehicle Level)

Control vehicle transferring between regions

Give precise position guidance within each region

Review of DeePC

Data-**e**nabl**e**d **P**redictive **C**ontrol Algorithm(DeePC)[1]

- Non-parametric method
- Use input/output trajectories

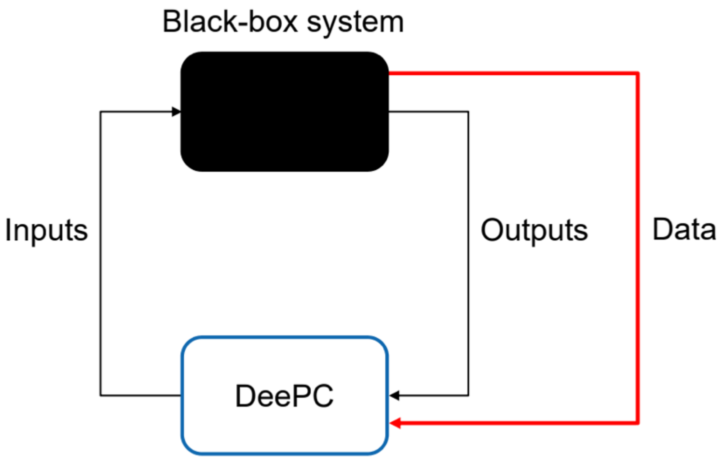
  Historical input and output data of length T:

  $$\begin{cases} u^{\mathrm{d}} = \mathrm{col}(u_1, ..., u_i, ..., u_T) \\ y^{\mathrm{d}} = \mathrm{col}(y_1, ..., y_i, ..., y_T) \end{cases}$$

  Hankel matrix:

  $$\mathcal{H}_L(u) := \begin{bmatrix} u_1 & u_2 & \cdots & u_{T-L+1} \\ u_2 & u_3 & \cdots & u_{T-L+2} \\ \vdots & \vdots & \ddots & \vdots \\ u_L & u_{L+1} & \cdots & u_T \end{bmatrix}$$

- Satisfy input/output constraints



DeePC for black-box system[2]

[1] Coulson et al., 2018, 'Data-Enabled Predictive Control: In the Shallows of the DeePC'.
[2] Huang et al., 2019, https://drive.google.com/file/d/1DU-JkNShGAX23ZKzGAcw97Qg_tLzVx-g/view

Model Predictive Control (MPC) v.s. DeePC

MPC:

$$\begin{aligned}
\underset{u,x,y}{\text{minimize}} \quad & \sum_{k=0}^{T_f-1} \left(\left\| y_k - r_{t+k} \right\|_Q^2 + \| u_k \|_R^2 \right) \\
\text{subject to} \quad & x_{k+1} = Ax_k + Bu_k, \ \forall k \in \{0, \ldots, T_f - 1\}, \\
& y_k = Cx_k + Du_k, \ \forall k \in \{0, \ldots, T_f - 1\}, \\
& x_{k+1} = Ax_k + Bu_k, \ \forall k \in \{-T_{\text{ini}}, \ldots, -1\}, \\
& y_k = Cx_k + Du_k, \ \forall k \in \{-T_{\text{ini}}, \ldots, -1\}, \\
& u_k \in \mathcal{U}, \ \forall k \in \{0, \ldots, T_f - 1\}, \\
& y_k \in \mathcal{Y}, \ \forall k \in \{0, \ldots, T_f - 1\}.
\end{aligned}$$

DeePC:

$$\begin{aligned}
\underset{g,u,y}{\text{minimize}} \quad & \sum_{k=0}^{T_f-1} \left(\left\| y_k - r_{t+k} \right\|_Q^2 + \| u_k \|_R^2 \right) \\
\text{subject to} \quad & \mathscr{H}_{T_{\text{ini}}+T_f} \begin{pmatrix} \hat{u} \\ \hat{y} \end{pmatrix} g = \begin{pmatrix} \hat{u}_{\text{ini}} \\ \hat{y}_{\text{ini}} \\ u \\ y \end{pmatrix}, \\
& u_k \in \mathcal{U}, \ \forall k \in \{0, \ldots, T_f - 1\}, \\
& y_k \in \mathcal{Y}, \ \forall k \in \{0, \ldots, T_f - 1\}.
\end{aligned}$$

Predictive model and state estimation in Model Predictive Control (MPC) is replaced by raw data in a Hankel matrix in DeePC[1].

[1] *Jeremy Coulson, John Lygeros, and Florian Dörfler. Sparse learning workshop 2020.*

Hankel Matrix

Historical input and output data of length T:

$$\begin{cases} u^{\mathrm{d}} = \mathrm{col}(u_1, \ldots, u_i, \ldots, u_T) \\ y^{\mathrm{d}} = \mathrm{col}(y_1, \ldots, y_i, \ldots, y_T) \end{cases}$$

Hankel matrix:

$$\mathcal{H}_L(u) := \begin{bmatrix} u_1 & u_2 & \cdots & u_{T-L+1} \\ u_2 & u_3 & \cdots & u_{T-L+2} \\ \vdots & \vdots & \ddots & \vdots \\ u_L & u_{L+1} & \cdots & u_T \end{bmatrix}$$

Definition 1 [1]: Persistent excitation
u is persistently exciting of order L if $H_L(u)$ has full row rank.

Lemma 1: If u^d is persistently exciting of order $T_{ini} + N + n$, then $col(u_{ini}, y_{ini}, u, y)$ is a trajectory of this system[1] if and only if there exists $g \in \mathbb{R}^{T-T_{ini}-N+1}$ such that

$$\begin{bmatrix} U^p \\ Y^p \\ U^f \\ Y^f \end{bmatrix} g = \begin{bmatrix} u_{ini} \\ y_{ini} \\ u \\ y \end{bmatrix}$$

Most recent measurement (of length T_{ini})

Future trajectory (Horizon N)

- "All trajectories can be reconstructed from finitely many, sufficiently rich previous trajectories"[2]

[1] Jan C. Willems, et al. "A note on persistency of excitation." Systems & Control Letters 54.4 (2005): 325-329.
[2] Jeremy Coulson, John Lygeros, and Florian Dörfler. Sparse learning workshop 2020.

Model Predictive Control (MPC) v.s. DeePC

MPC:

$$
\begin{aligned}
\underset{u,x,y}{\text{minimize}} \quad & \sum_{k=0}^{T_f-1} \left(\left\| y_k - r_{t+k} \right\|_Q^2 + \|u_k\|_R^2 \right) \\
\text{subject to} \quad & x_{k+1} = Ax_k + Bu_k, \ \forall k \in \{0,\dots,T_f-1\}, \\
& y_k = Cx_k + Du_k, \ \forall k \in \{0,\dots,T_f-1\}, \\
& x_{k+1} = Ax_k + Bu_k, \ \forall k \in \{-T_{\text{ini}},\dots,-1\}, \\
& y_k = Cx_k + Du_k, \ \forall k \in \{-T_{\text{ini}},\dots,-1\}, \\
& u_k \in \mathcal{U}, \ \forall k \in \{0,\dots,T_f-1\}, \\
& y_k \in \mathcal{Y}, \ \forall k \in \{0,\dots,T_f-1\}.
\end{aligned}
$$

System model

DeePC:

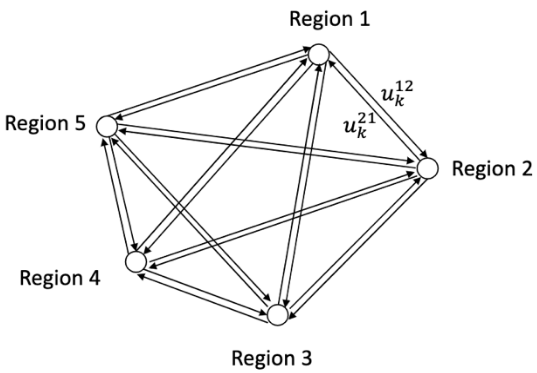
$$
\begin{aligned}
\underset{g,u,y}{\text{minimize}} \quad & \sum_{k=0}^{T_f-1} \left(\left\| y_k - r_{t+k} \right\|_Q^2 + \|u_k\|_R^2 \right) \\
\text{subject to} \quad & \mathcal{H}_{T_{\text{ini}}+T_f}\begin{pmatrix}\hat{u} \\ \hat{y}\end{pmatrix} g = \begin{pmatrix}\hat{u}_{\text{ini}} \\ \hat{y}_{\text{ini}} \\ u \\ y\end{pmatrix}, \\
& u_k \in \mathcal{U}, \ \forall k \in \{0,\dots,T_f-1\}, \\
& y_k \in \mathcal{Y}, \ \forall k \in \{0,\dots,T_f-1\}.
\end{aligned}
$$

Hankel Matrix

Predictive model and state estimation in Model Predictive Control (MPC) is replaced by raw data in a Hankel matrix in DeePC[1].

[1] Jeremy Coulson, John Lygeros, and Florian Dörfler. Sparse learning workshop 2020.

Problem Formulation



Assume AMoD system is a LTI system with measurable disturbance.

Control input

u_{IJ} : # of empty vehicles be relocated from region I to region J.

Output

y_I : # of successfully answered requests in region I.

External measurable disturbance

w_I^o : # of requests which starts from region I,
w_I^D : # of requests which ends in region I.

Problem Formulation

Pengbo ZHU

$$f(u_{k+i}, y_{k+i}) = \sum_{i=0}^{N-1} -\|Qy_{k+i}\| + \|Ru_{k+i}\|$$

1. Answer more requests

2. Consider rebalancing cost

$$\min_{g,u,y,\sigma_y} f(u_{k+i}, y_{k+i}) + \lambda_g \|g\|_I^2 + \lambda_y \|\sigma_y\|_I^2$$

$$subject\quad to \quad \begin{bmatrix} U^p \\ W^p \\ Y^p \\ U^f \\ W^f \\ Y^f \end{bmatrix} g = \begin{bmatrix} u_{ini} \\ w_{ini} \\ y_{ini} \\ u \\ w \\ y \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \sigma_y \\ 0 \\ 0 \\ 0 \end{bmatrix},$$

Slack variables & Regularization
1. Feasibility
2. Avoid overfitting

Most recent measurement:
Updated at every step

Hankel matrix:
Constructed by historical data

$$u_{k+i} \geq 0, \forall i \in 0, 1, \ldots, N-1.$$

$$\sum_{J=1}^{K_n} u_k^{IJ} = n_k^I, \forall I \in 0, 1, \ldots, K_n.$$

$$y_{k+i} \geq 0, \forall i \in 0, 1, \ldots, N-1.$$

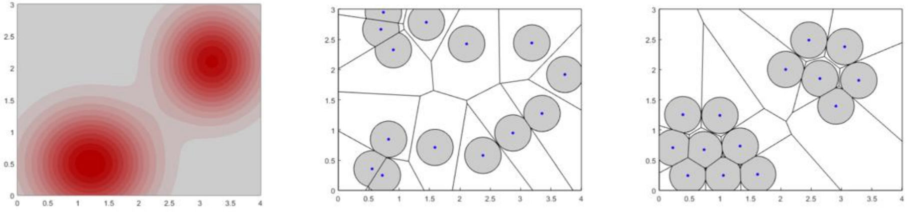
n_k^I: # of empty vehicles in Region I

Constraints:
1. Positivity
2. Operate based on current availability

Lower-level Controller

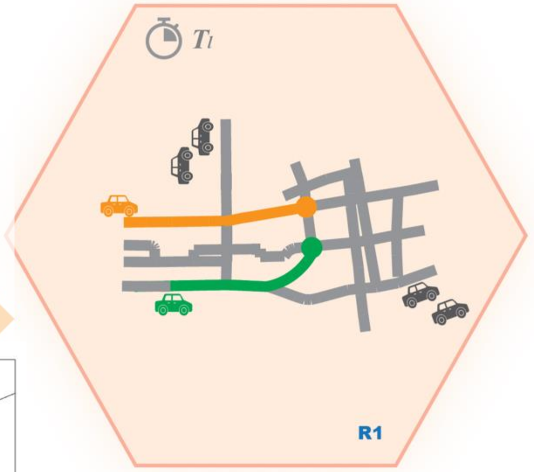
Coverage Control

[1] **P. Zhu,** G. Ferrari-Trecate, and N. Geroliminis,
IEEE Transactions on Control Systems Technology, 2024;



(a) Demand density distribution (b) Initial configuration of agents (c) Optimal configuration of agents

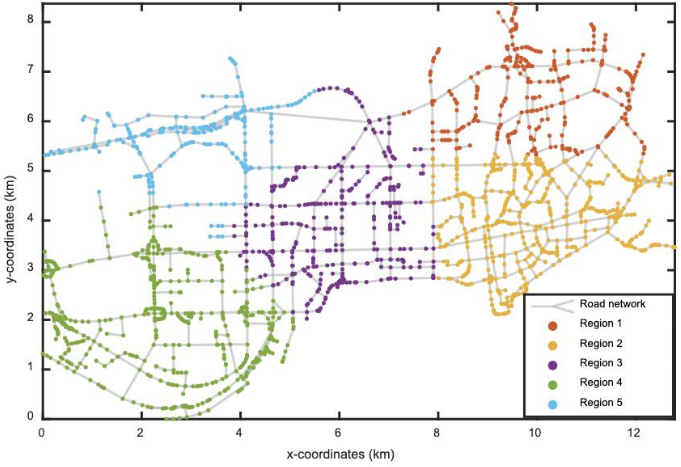
Align the vehicle distribution with the demand density



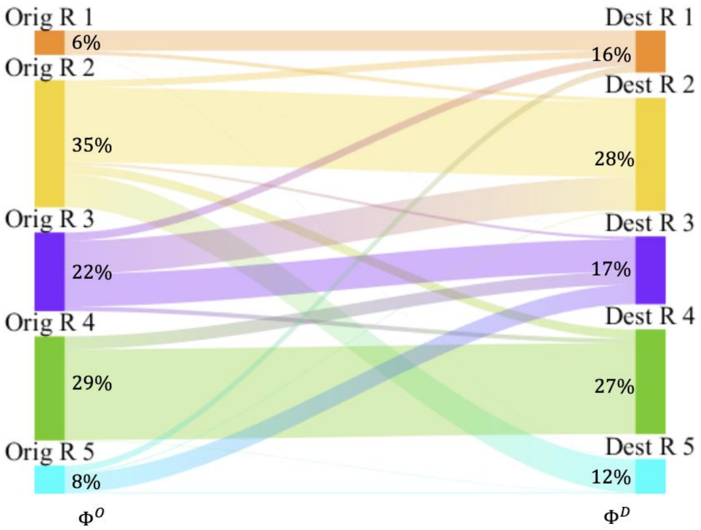
Low Level (Vehicle Level)

Pengbo ZHU

Experimental Setting



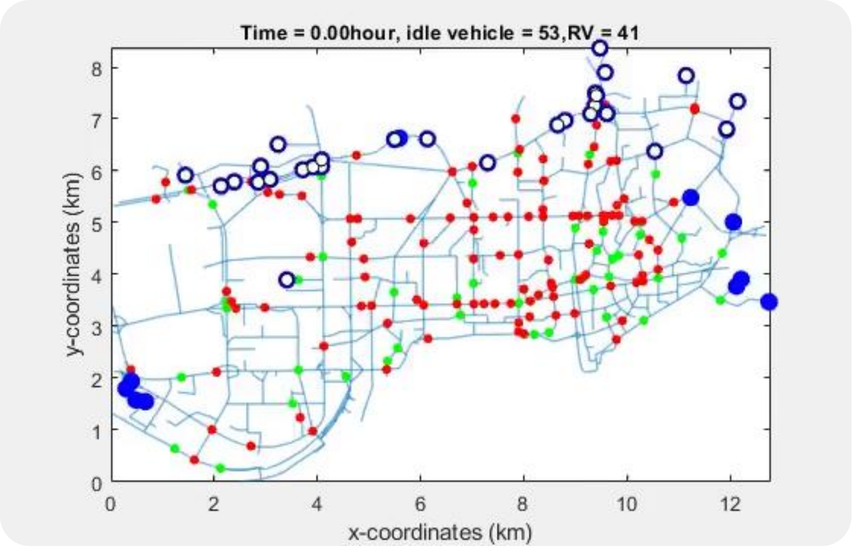
City area is clustered into 5 regions

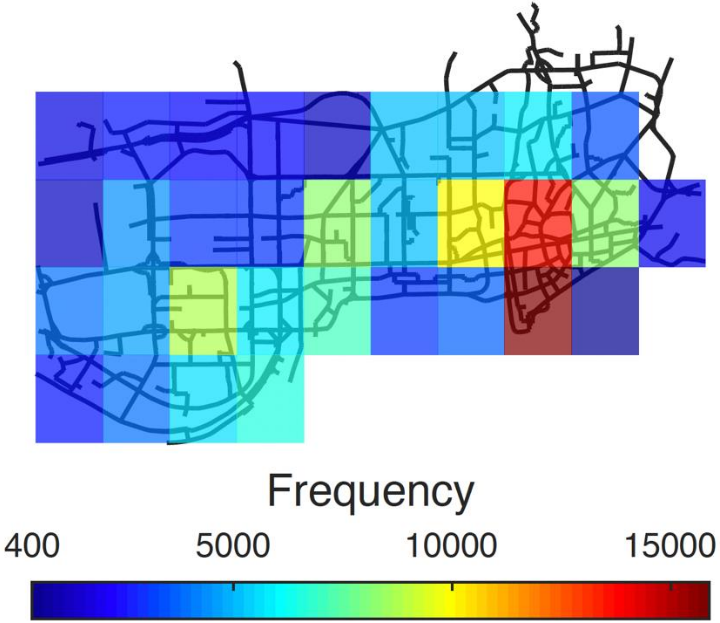


Trip distribution from Region I to Region J.

- Regions 1 and 5 have less demand than Regions 2, 3, and 4.
- Without intervention, empty vehicles will accumulate in low-demand areas.

Case Study





Blue circles (High): vehicles rebalanced to another region
Blue dots (Low):   vacant vehicles staying in current region
green: passenger-assigned vehicle,
red: passenger-carrying vehicle.

Trip Origin Distribution

Results

- 3-hour simulation
- 5500 orders
- Fleet size = 300

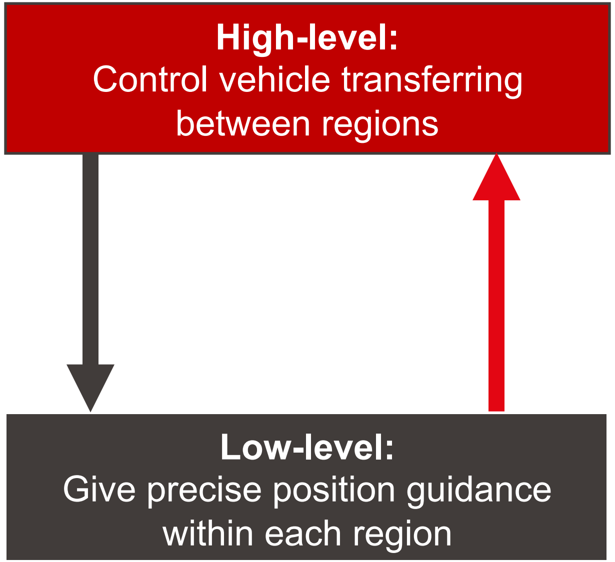
	Answer rate (%)	Average waiting time (s)	Rebalancing distance (km)
Upper + Lower	**88.5**	**134.7**	**4276.7**
Upper Only	85.5	139.3	4958.4
Lower Only	84.2	139.0	5155.1
No Control	53.8	155.6	0.0
No Control (fleet size = 450)	69.2	140.4	0.0

NB:
1. Upper + Lower: apply DeePC as high-level controller, meanwhile, apply coverage control for each region.
2. Upper Only: only apply DeePC for the inter-regional vehicle transfer.
3. Lower Only: only apply coverage control for the whole map.
4. No Control: vehicles stop at their last destinations, until they are matched with passengers again.

Pengbo ZHU

Summary

- **Hierarchical control** structure bridges between **macroscopic** and **microscopic** scopes.
- **Scalability + Modularity**



- Data-driven control method can learn how to control the system directly from the collected data, bypassing the process to build a model/identify the parameters.
- Coverage control-based method yields its advantage on reducing waiting time.

Part II:

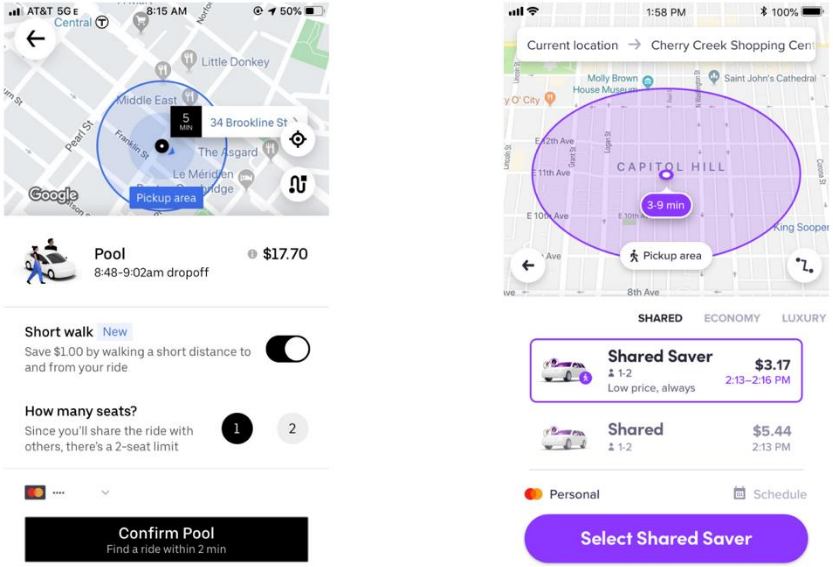
Detour planning for Ride-pooling Vehicles

Pengbo ZHU

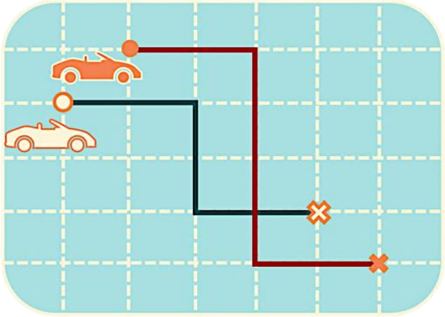
Background

Ride-pooling systems

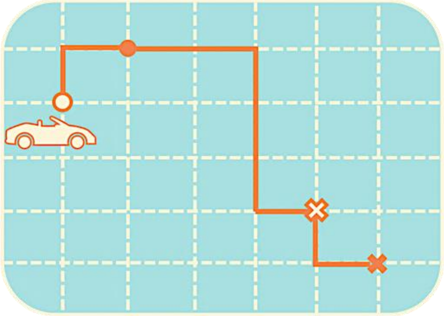
- UberPool and Lyft gives passengers the option to **share their ride** for a more **affordable price**.



Pooling services by UberPool and LyftShare[1]



Two solo trips



One pooled trip

[1] *Zhang et al., 2023, 'Routing Optimization with Vehicle-Customer Coordination'*

Problem Description

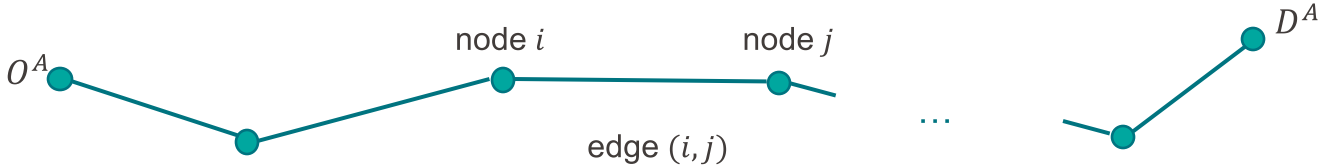
Literature Review

- Matching/dispatching schemes between passengers, between passenger and vehicles, etc.,
- Service Plan for already-matched passengers,
- Route Choice among pre-defined routes/paths…

However, we focus on:

1) Plan a path for any origin and destination nodes on a graph.

Path = a sequence of edges.

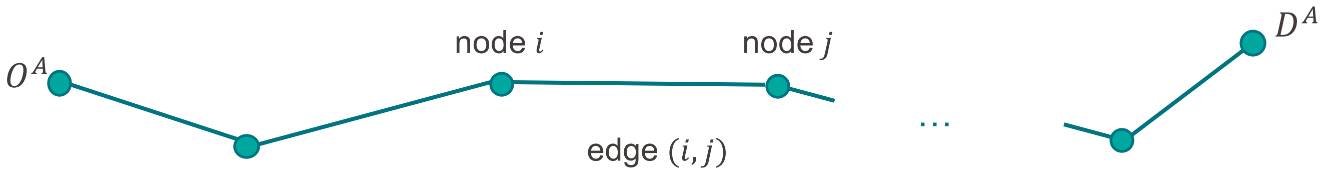


Problem Description

However, we focus on:

1) Plan a path for any origin and destination nodes on a graph.

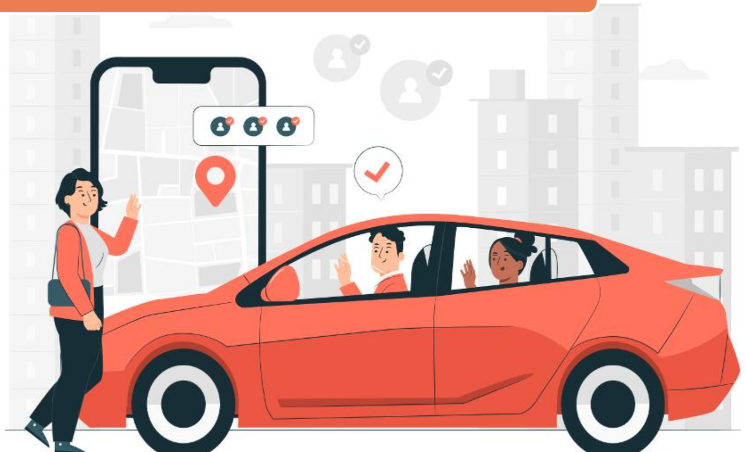
Path = a sequence of edges.



2) for Partially-occupied Vehicles (PoVs)

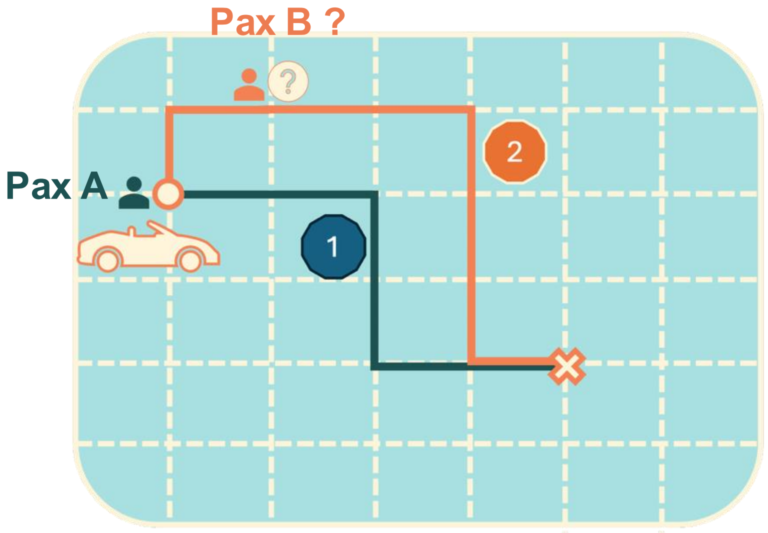
—— **non-empty vehicles with available seats**

A taxi already has passengers on:
It has “**must go**” destinations,
but can contribute to mobility accessibility.



Detour Planning for Partially-occupied Vehicles (PoVs)

Pengbo ZHU



Detour Planning

For the current Pax A,

Plan a path from O^A (origin) to D^A (destination)

1) Attractiveness: higher possibility to pick up a second passenger en route

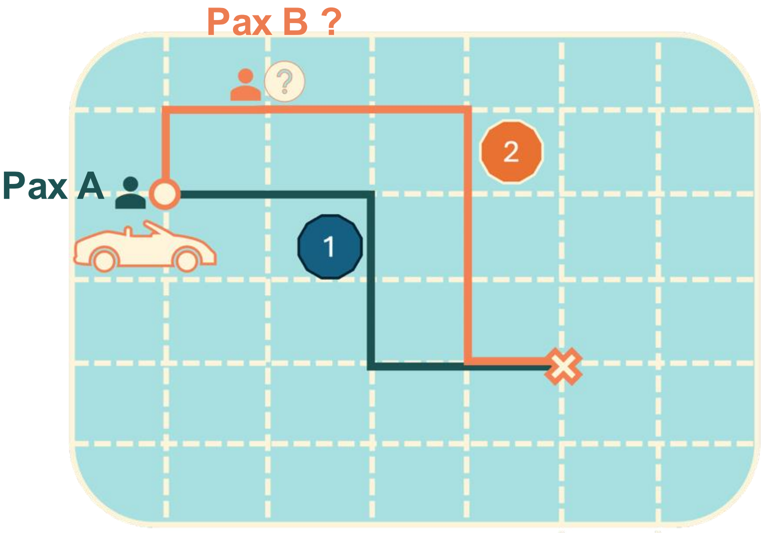
2) Repulsiveness: coordinate with other vehicles

1 Directly dropping off Pax A

2 A planned detour path

Detour Planning for Partially-occupied Vehicles (PoVs)

Pengbo ZHU



Detour Planning

1 Directly dropping off Pax A

2 A planned detour path

For the current Pax A,

Plan a path from O^A (origin) to D^A (destination)

1) **Attractiveness:** higher possibility to pick up a second passenger en route

2) **Repulsiveness:** coordinate with other vehicles

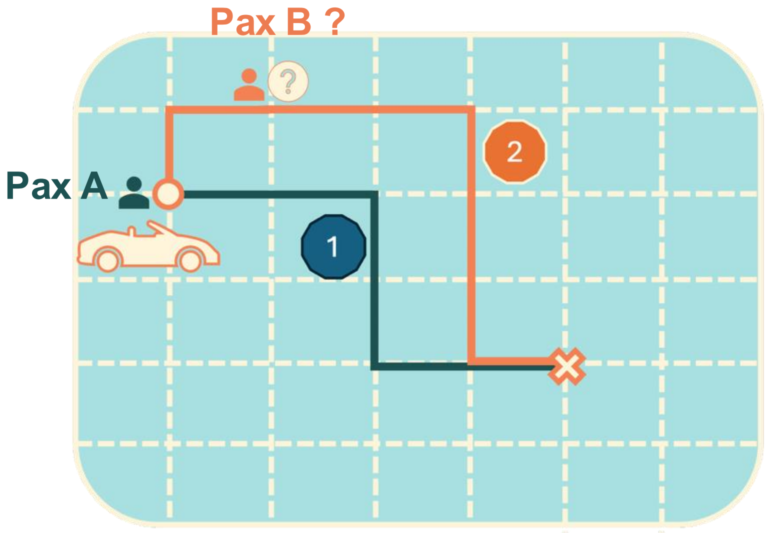
Modeling: The matching probability is

$$p_{ij}(t) = 1 - \zeta e^{-\frac{\lambda_{ij}(t)}{\eta n_{ij}(t)}}$$

Attractiveness: Poisson arrival rate of customers

Repulsiveness: No. of available vehicles

Detour Planning for Partially-occupied Vehicles (PoVs)



Detour Planning

1 Directly dropping off Pax A

2 A planned detour path

For the current Pax A,

Plan a path from O^A (origin) to D^A (destination)

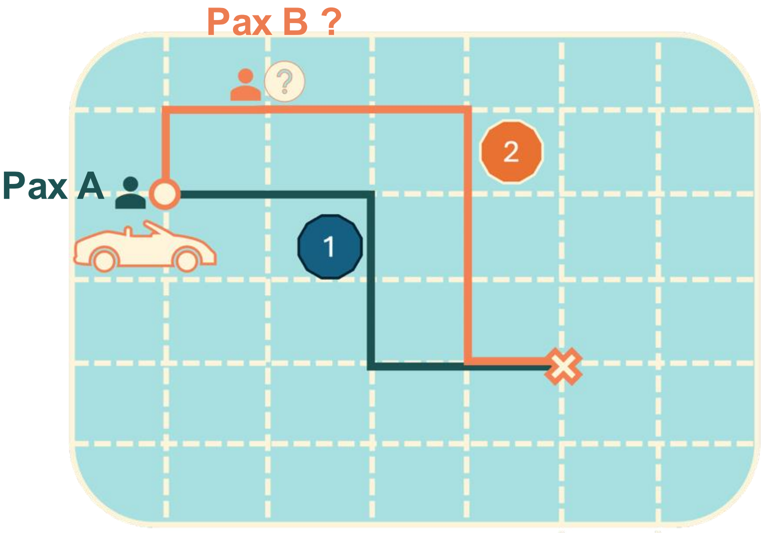
1) Attractiveness: higher possibility to pick up a second passenger en route

2) Repulsiveness: coordinate with other vehicles

3) Constraints: not to exceed the max. detour time

- If it **matches up with Pax B, pool the trip;**
- If it fails, it ends up as a solo trip.

Detour Planning for Partially-occupied Vehicles (PoVs)



Detour Planning

1 Directly dropping off Pax A

2 A planned detour path

For the current Pax A,

Plan a path from O^A (origin) to D^A (destination)

1) Attractiveness: higher possibility to pick up a second passenger en route

2) Repulsiveness: coordinate with other vehicles

3) Constraints: not to exceed the max. detour time

- If it **matches up with Pax B, pool the trip;**
- If it fails, it ends up as a solo trip.

4) Revenue: Compatible & Profitable

Problem Formulation

Decision Variable

$$x_{ij} = \begin{cases} 1 & if\ edge\ (i,j)\ is\ part\ of\ the\ path \\ 0 & otherwise \end{cases}$$

$0 \leq P_i^{no} \leq 1$, a continuous variable,
indicating whether the vehicle has not picked up any passenger before traversing node i.

Objective Function

maximize the **expected revenue** from successfully matching with a Pax B along the chosen path:

$$\max \sum_{(i,j) \in E} r_{ij} \cdot p_{ij} \cdot P_i^{no} \cdot x_{ij}$$

(A) Average potential revenue from a match (B) Pool-match probability

Problem Formulation

Pengbo ZHU

Objective Function

maximize the **expected revenue** from successfully matching with a Pax B along the chosen path:

$$\max \sum_{(i,j)\in E} r_{ij} \cdot p_{ij} \cdot P_i^{no} \cdot x_{ij}$$

subject to

$$\sum_{j:(O^A,j)\in E} x_{O^A j} - \sum_{j:(j,O^A)\in E} x_{j O^A} = 1,$$

$$\sum_{j:(i,j)\in E} x_{ij} - \sum_{j:(j,i)\in E} x_{ji} = 0$$

$$\sum_{j:(D^A,j)\in E} x_{D^A j} - \sum_{j:(j,D^A)\in E} x_{j D^A} = -1$$

$$u_{O^A} = 1; u_i - u_j + |Q| \cdot x_{ij} \leq |Q| - 1$$

$$2 \leq u_i \leq |Q|$$

$$P_{O^A}^{no} = 1$$

$$Q_{ij}, \leq P_i^{no} Q_{ij} \leq x_{ij}, Q_{ij}, \geq P_i^{no} - (1 - x_{ij}) Q_{ij} \geq 0$$

$$P_j^{no} = \sum_{i:(i,j)\in E} Q_{ij} \cdot (1 - p_{ij})$$

$$\sum_{(i,j)\in E} x_{ij} \cdot \omega_{ij} \leq L_{\max}$$

(Start node and end node)

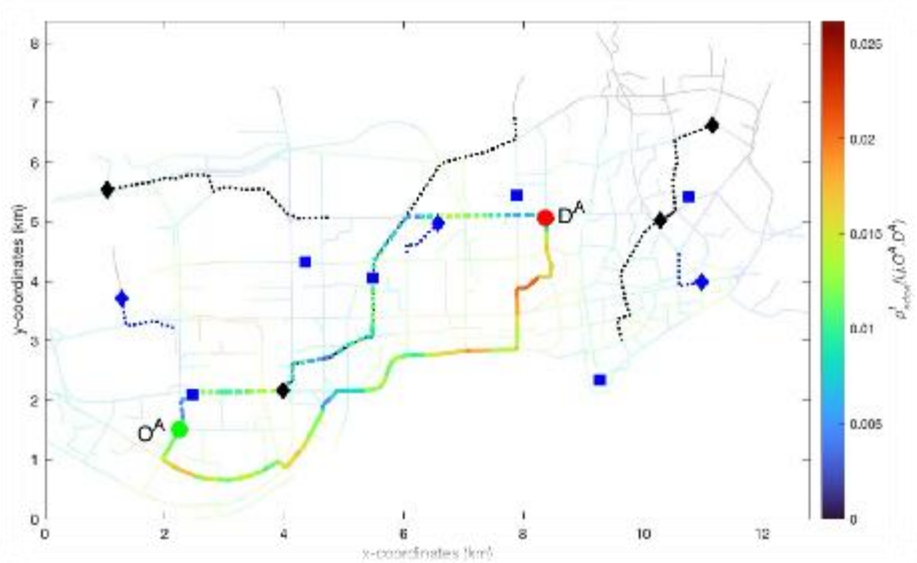
(Continuity in path)

(Cycle prevention)

(Pooling-Match can happen only once)

(Total maximum detour for Pax A)

Case Study



Blue Squares: Empty vehicles
Blue diamonds: PoVs
Blue dotted lines: planned routes of PoVs
Black: dropping off their last passenger

Shortest (Travel time Only): 8.83 km
Proposed: 10.39 km

Our proposed method designs a path that:

- ✓ Avoid 'overlapping' with other planned routes,
- ✓ Keep distance from other empty vehicles,
- ✓ Pass through higher demand area,
- ✓ An acceptable detour distance.

Results

Fleet size = 100;
 # of all order: 1700 issued in 3 hours

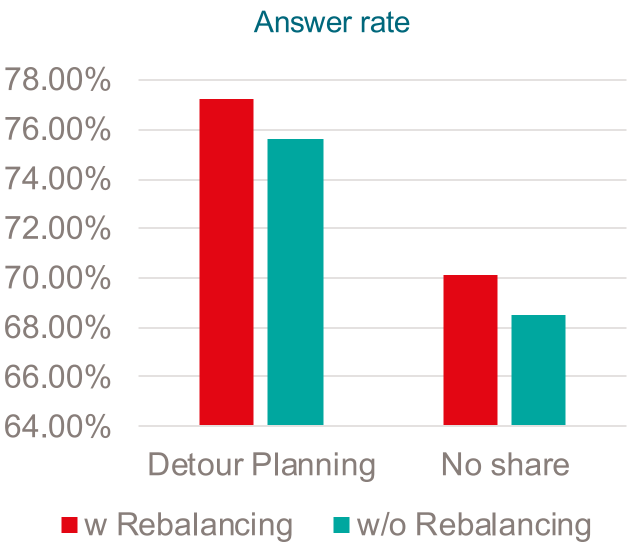
Pooling: $f_B^p = 2.0\ USD, f_p^p = 0.8\ USD/km$
Discount: 20% off

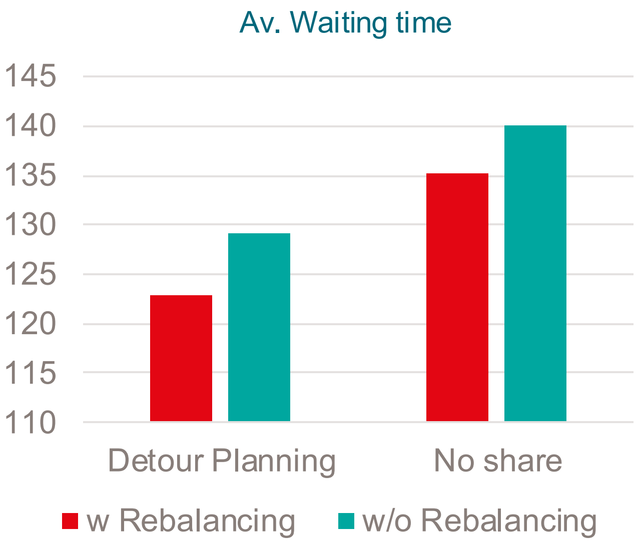
Solo: $f_A^s = 2.2\ USD,\ f_p^s = 1.0\ \mathrm{USD/km}$

	Revenue ($)	Answer rate (%)	Av. waiting time (s)	No. of shared
Proposed	**1850.0**	**75.6**	**129.1**	**722**
Shortest	1667.4	72.1	133.8	501
No share	1772.1	68.5	140.0	0

EPFL

Detour Planning + Idle Vehicle Rebalancing?





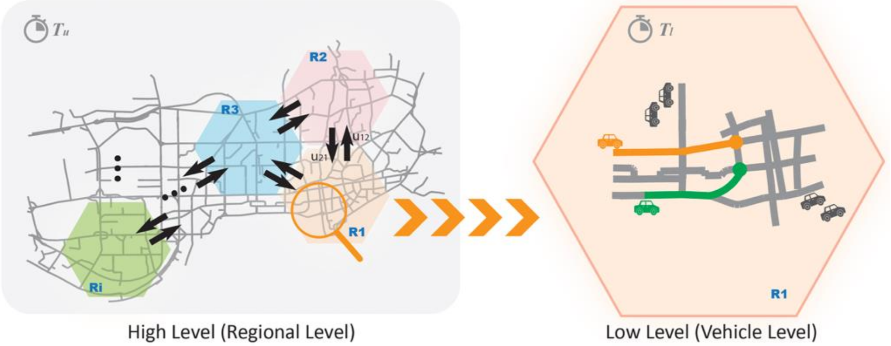
Pengbo ZHU

Pengbo ZHU

Take home message

Part I: Empty Vehicle Repositioning

Where should an empty Uber go?



Part II: Detour Planning for Ride-Pooling Vehicles

Which path a pooling-vehicle should traverse?