"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

Challenges in the field of research

Complex interactions

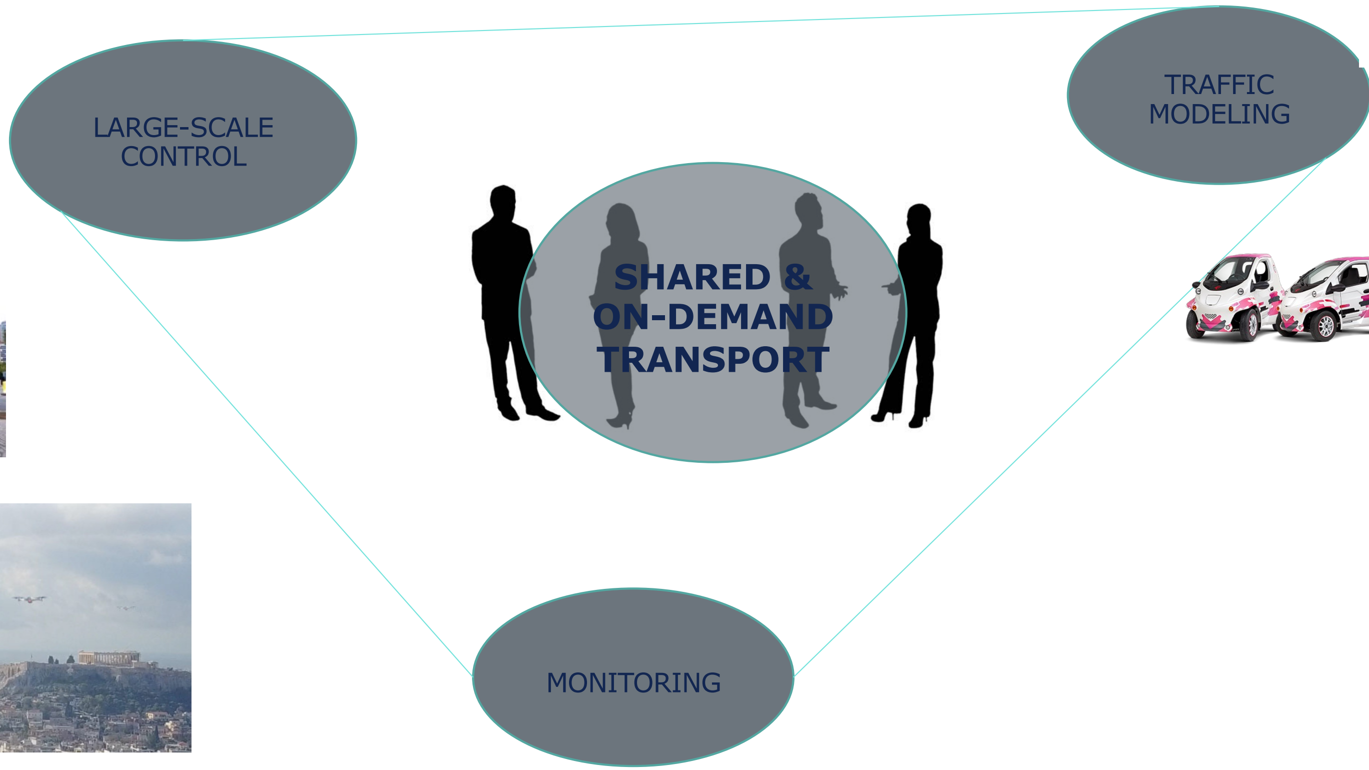
- Multimodality
- Components adapt
- Competing operators
- Spreading phenomena

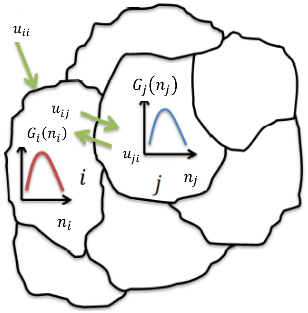
Limits to predictability

Limits to centralization

Limits to cooperation

A human-centric ubiquitous mobility





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

Introduction

weekday vehicle circulation patterns in Austin, Texas

Teal circles : areas that net attract vehicles;

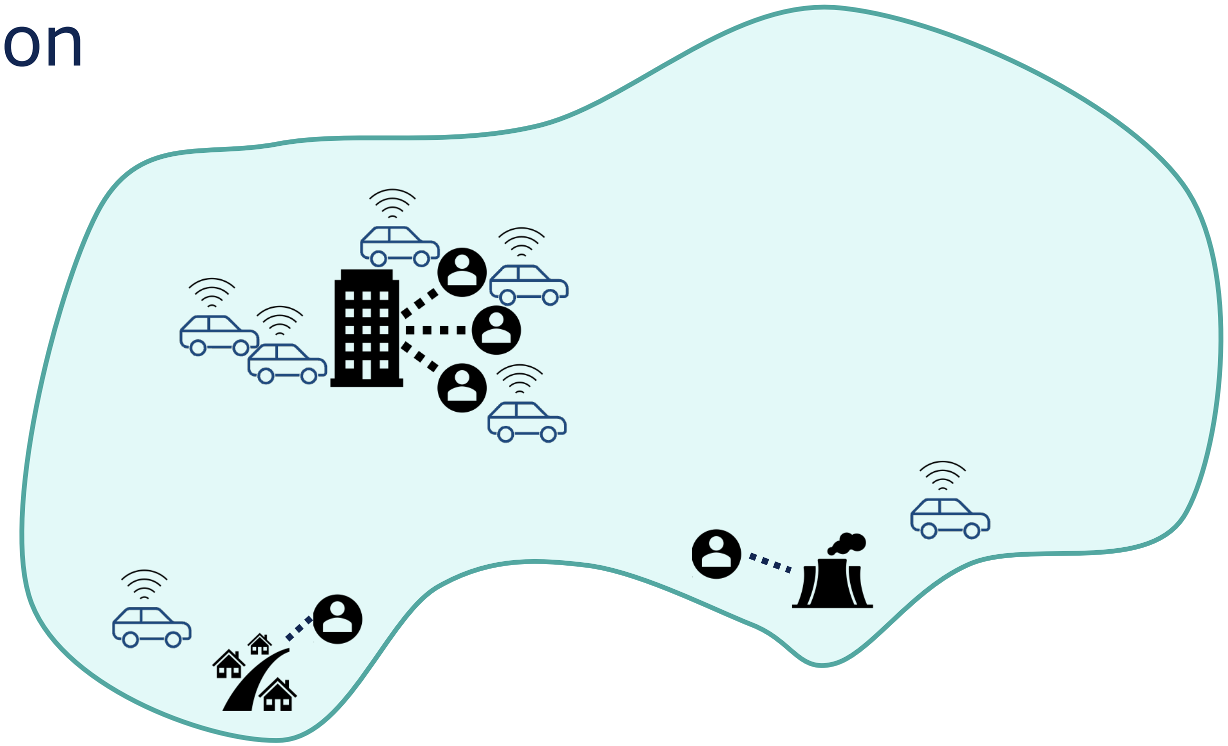
Red circles : areas that net lose vehicles.

Imbalance between supply and demand:

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

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

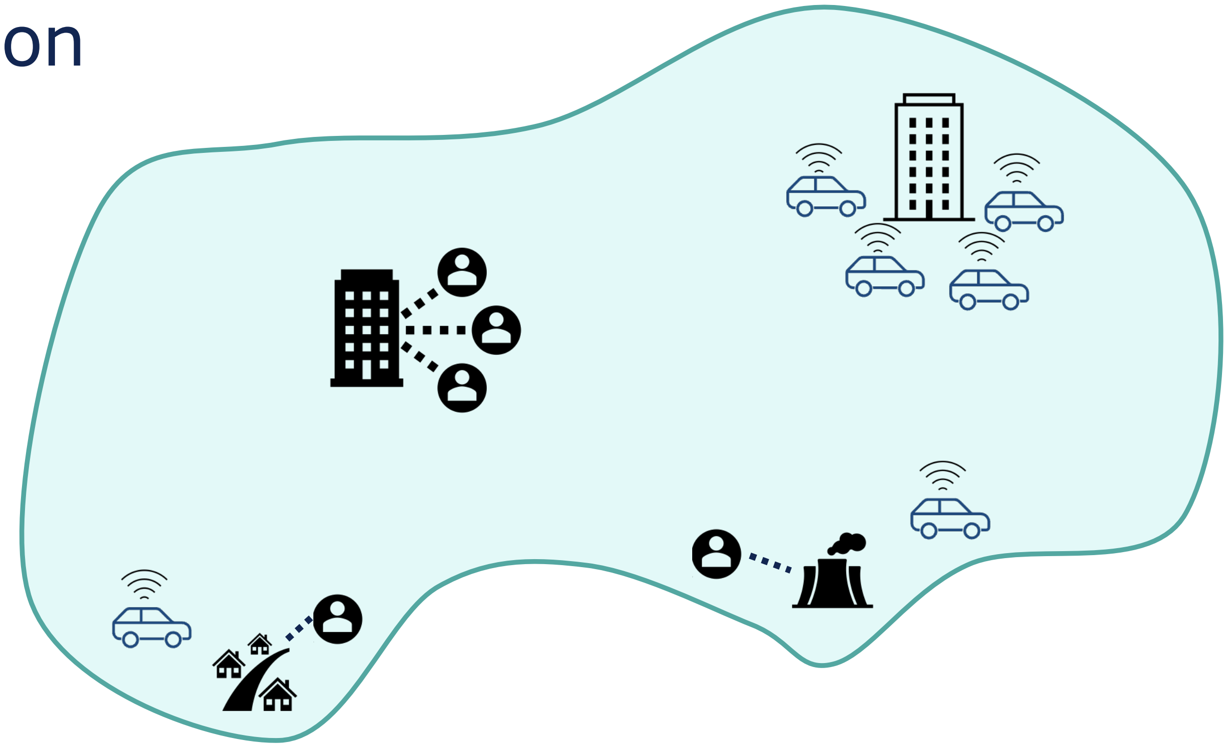
Motivation



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.

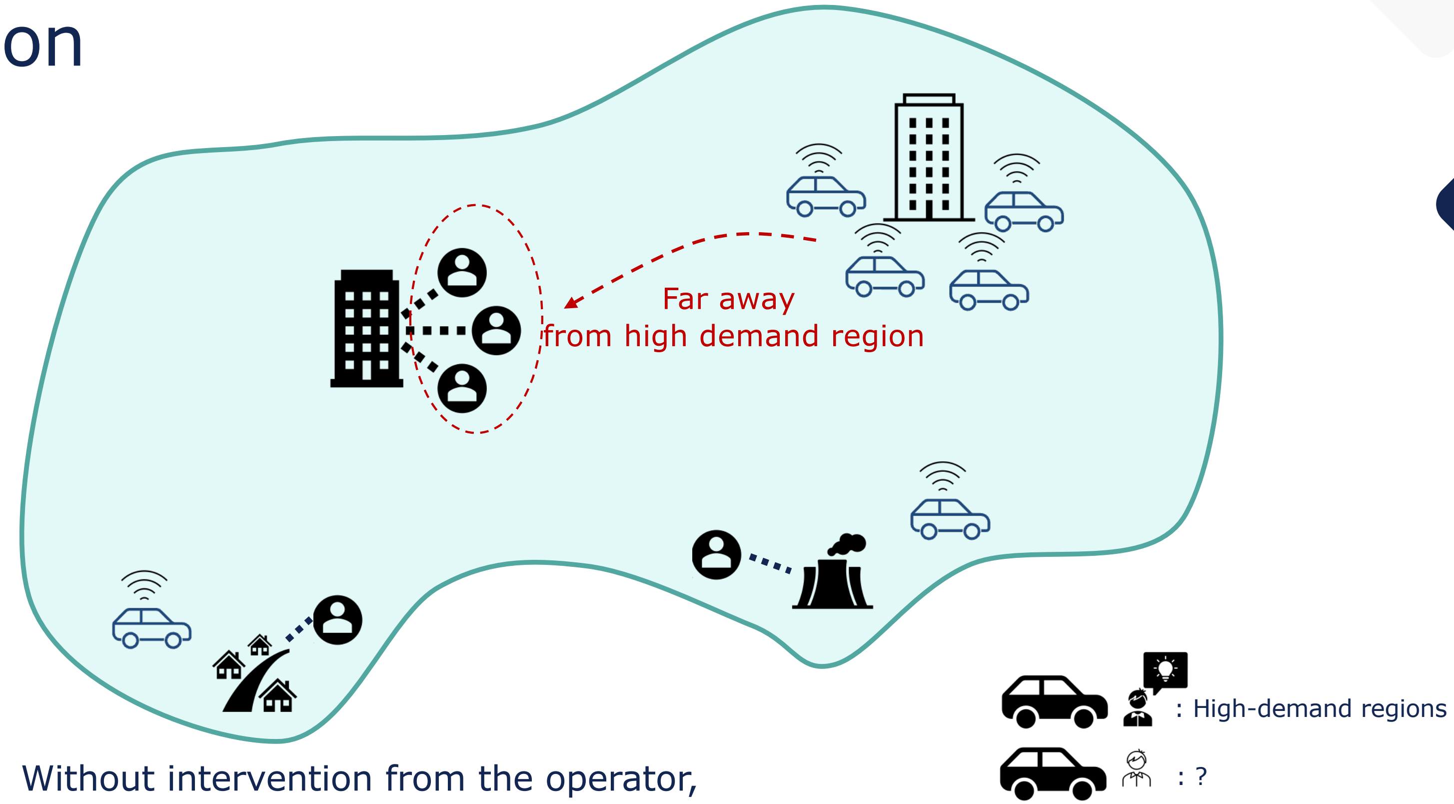
Motivation



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,

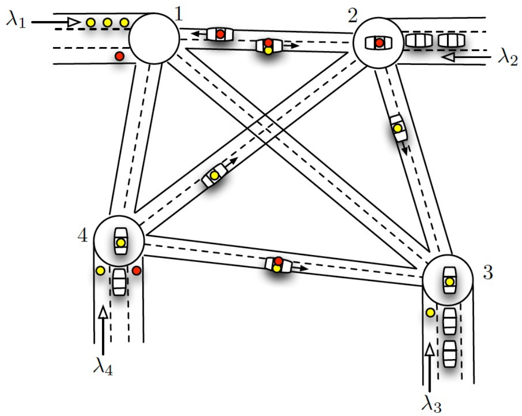
The fleet will be increasingly concentrated in low-demand locations, rather than high-demand locations.

Goal: rebalancing vehicles

- Relocating vehicles to the high-demand regions.

State-of-the-art

- applied to car-sharing or bike-sharing systems but few in the ride-sourcing systems.
- operated with a long rebalancing interval
- limited to several stations with a small fleet size.



Region/Station-level rebalancing strategy

Q: Is it possible to get a more precise position guidance?

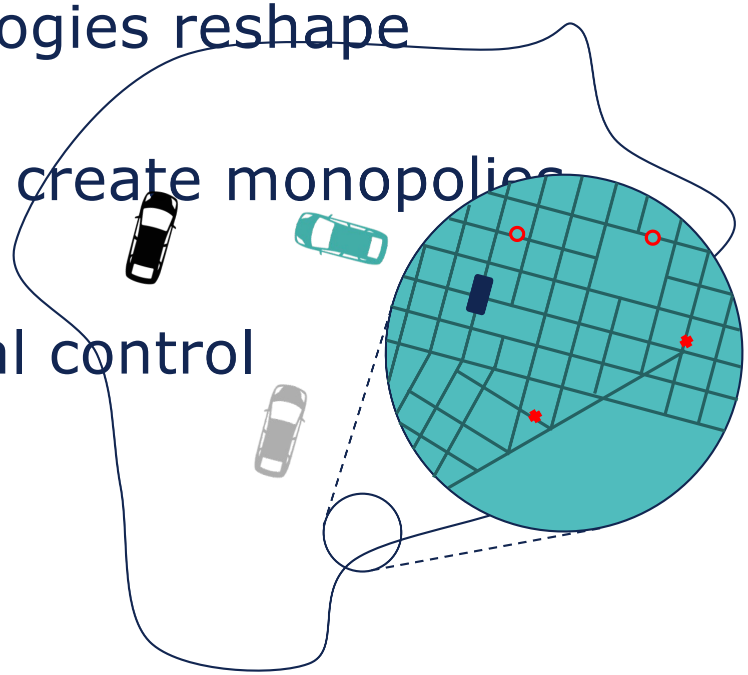
1. https://www.eltis.org/discover/news/stockholms-bike-sharing-scheme-receive-electrifying-upgrade-sweden
2. [Smith,2013]. "Rebalancing the rebalancers: Optimally routing vehicles and drivers in mobility-on-demand systems."

NCCR Automation

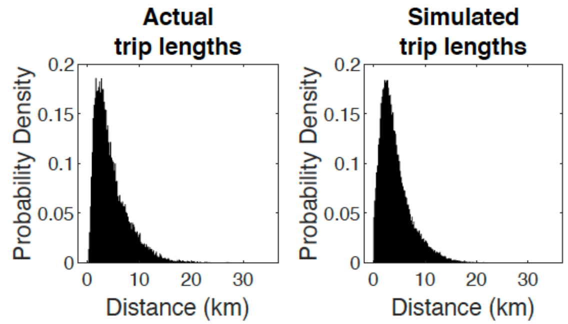
SHARED & ON-DEMAND TRANSPORT

Vision

- A new mobility paradigm through flexibility, cooperation and collectivity.
- Revolutionize how emerging technologies reshape mobility
- Avoid erratic developments that can create monopolies
- Identify new ways of governance
- Integrate distributed and hierarchical control

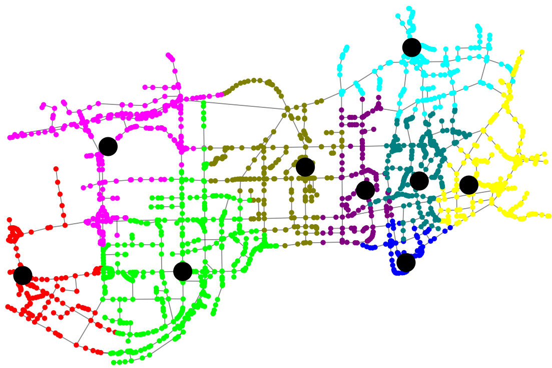


Data and simulation description



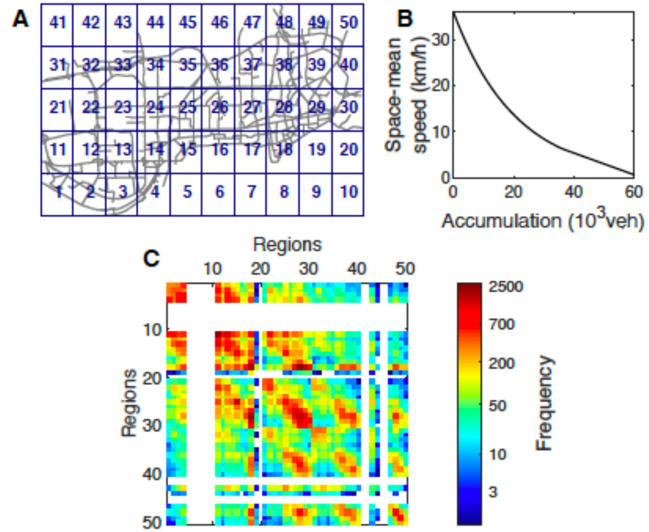
- Trip-based MFD model

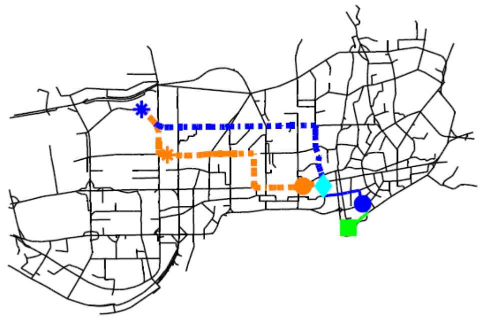
$$l_0 = \int_{t_0}^{t_0+\tau_0} v(n(k))\, dk$$



Locations of depots and closest nodes.

- Fleet size
- Willingness to share
- Drivers' behavior
  - Cruise immediately
  - Rebalance to depots;





Greedy Pooling policy

O-D demand

Beojone and Geroliminis, TR part C, 2021

Results

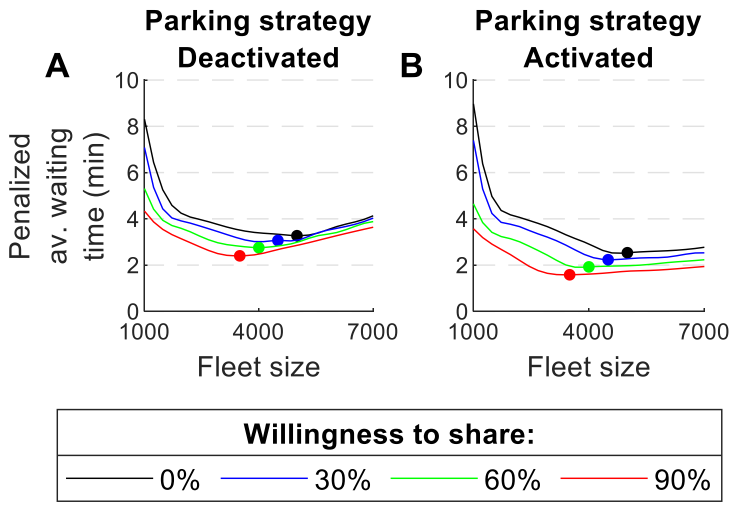


Figure 1. Average Waiting times.

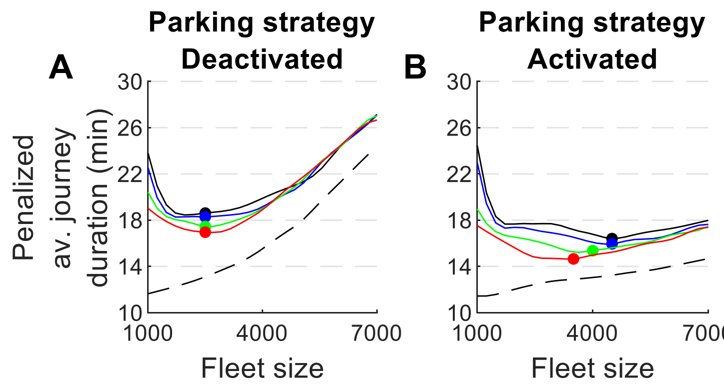


Figure 2. Average journey duration.

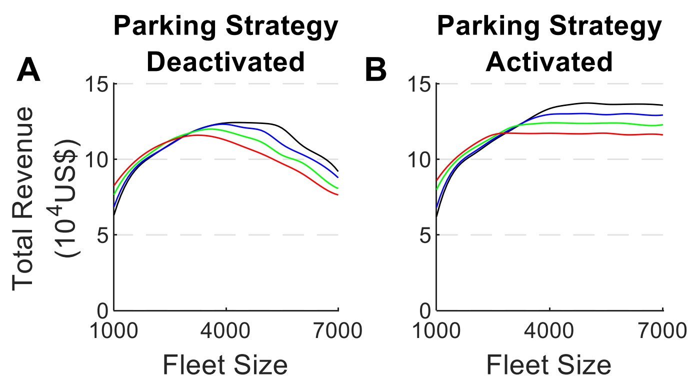
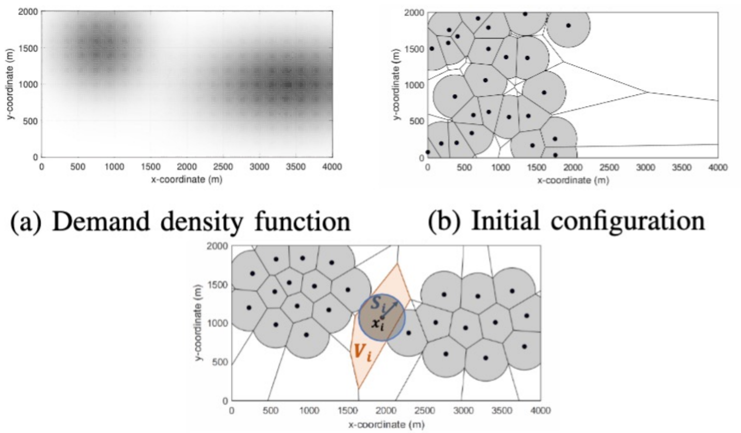


Figure 3. System's revenues.

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



(a) Demand density function (b) Initial configuration

(c) Final configuration

Fig. 1: Continuous case study

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

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

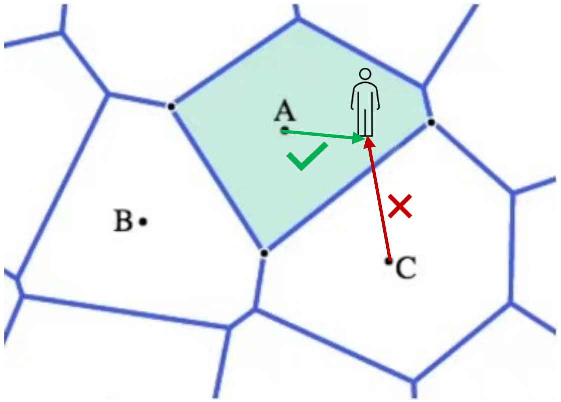
$\varphi(q)$: demand density function,

[1] P. Zhu et al, "Idle-vehicle Rebalancing Coverage Control for Ride-sourcing systems," ECC 2022.

Problem Formulation

Voronoi partition:

- The partitioning of a plane with n points into covex polygons.
- Each cell contains one generator/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

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

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

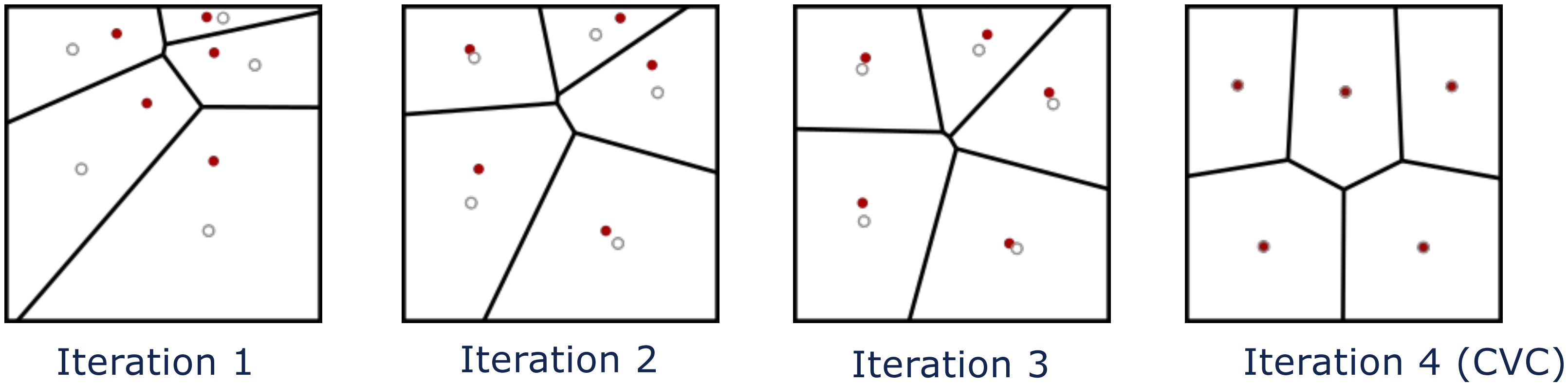
where $\varphi(q)$ is the demand density function.

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

[1] Du, Q., V. Faber and M. Gunzburger (1999) *Centroidal voronoi tessellations: Applications and algorithms*

Lloyd’s algorithm:

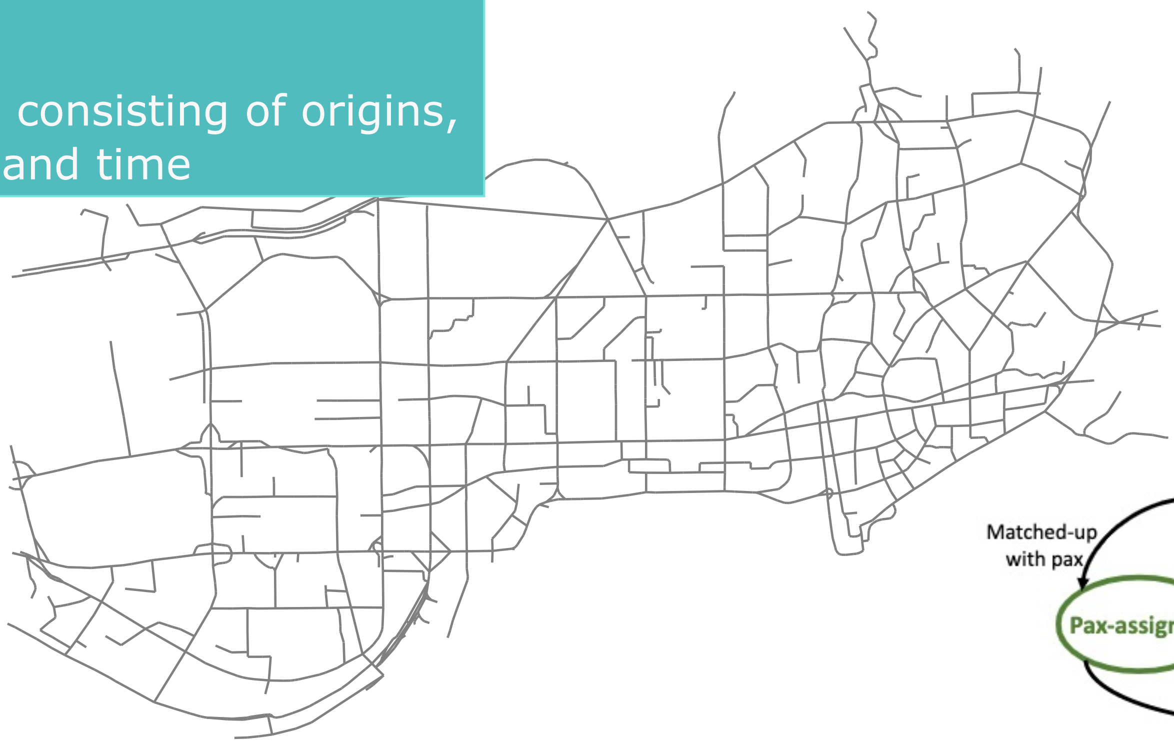
1) Obtain the Voronoi Parition;
2) Calculate the centroid of each cell;
3) Move from current position to the centroid;
4) Return to Step 1), until find the centroidal Voronoi configuration (CVC).

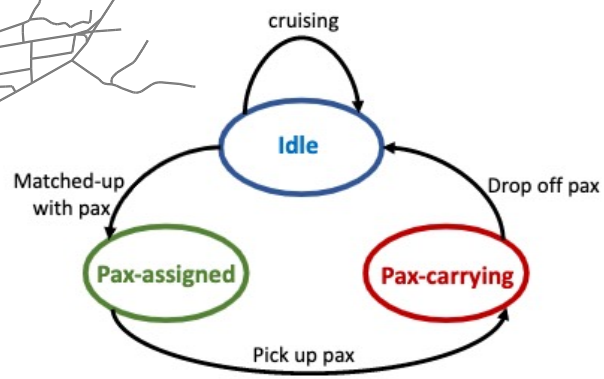


Example of Lloyd's algorithm.
The Voronoi diagram of the current positions (red) at each iteration is shown.
The gray circles denote the centroids of the Voronoi cells

Case Study : Shenzhen, China

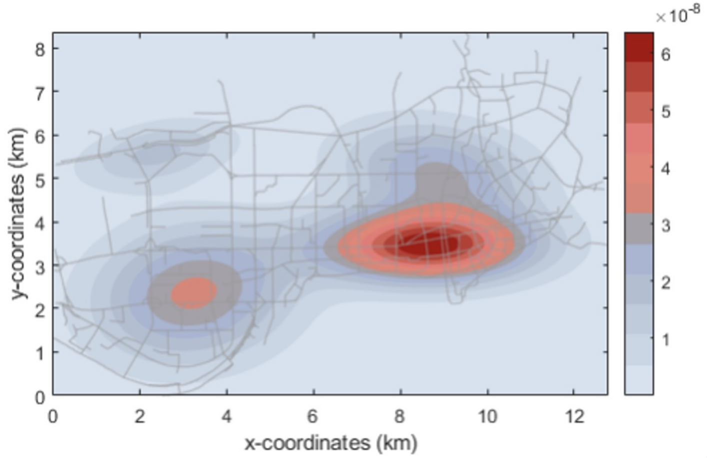
Simulated network:
- Shenzhen, China
- 1858 nodes
- 2013 links
- 199,819 trips consisting of origins, destinations, and time



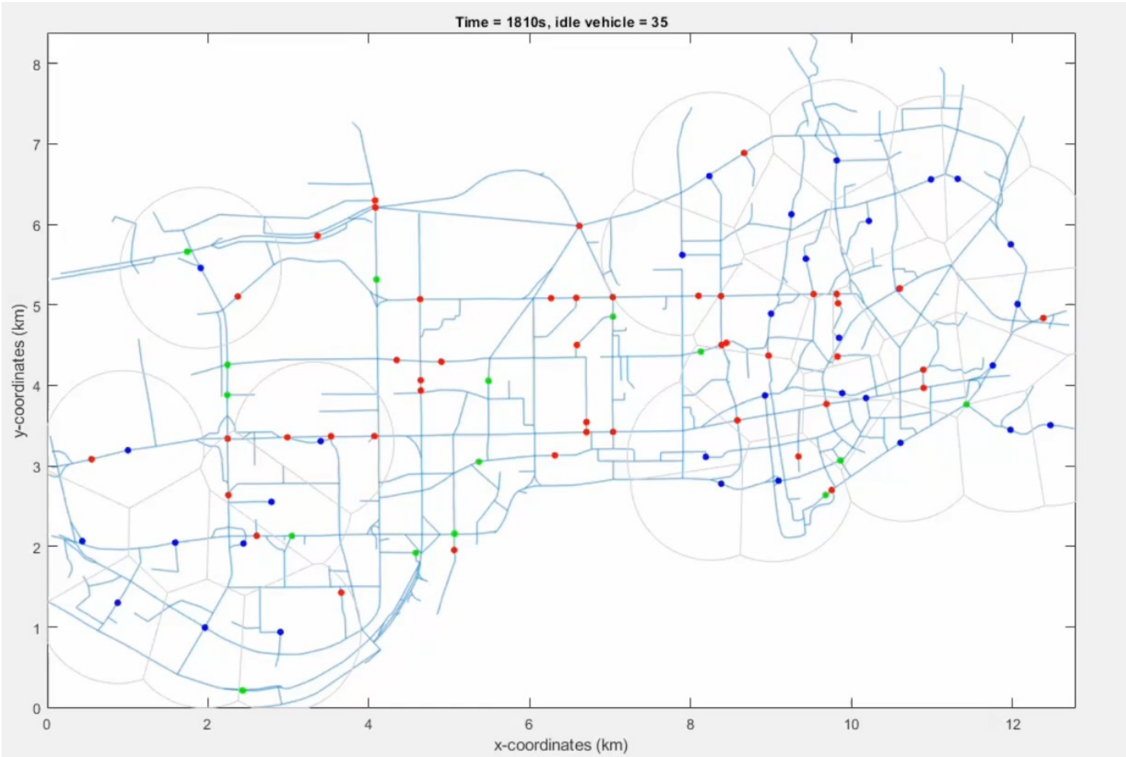


[1] Beojone, Caio & Geroliminis, Nikolas. (2020). *On the inefficiency of ride-sourcing services towards urban congestion.*

Case Study: Shenzhen, China



Demand density function



Demo video

blue: idle vehicle (i.e., empty, looking for a passenger),

green: passenger-assigned vehicle,

red: passenger-carrying vehicle.

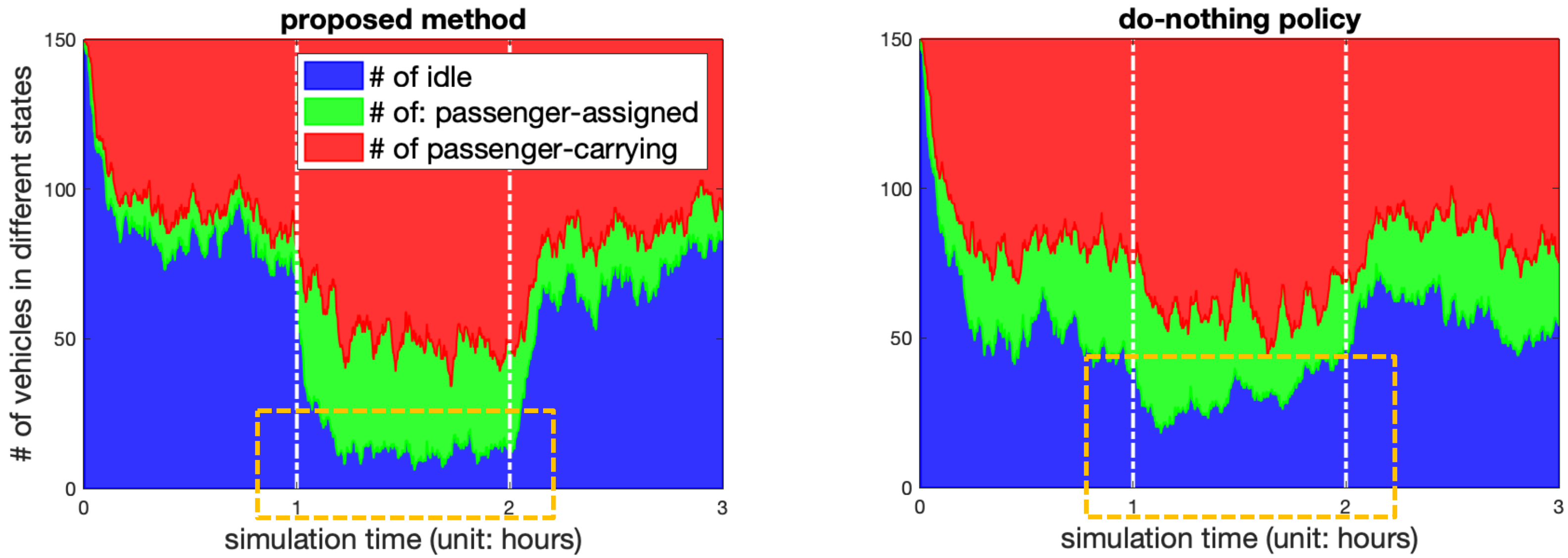
Results and Analysis

- 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.

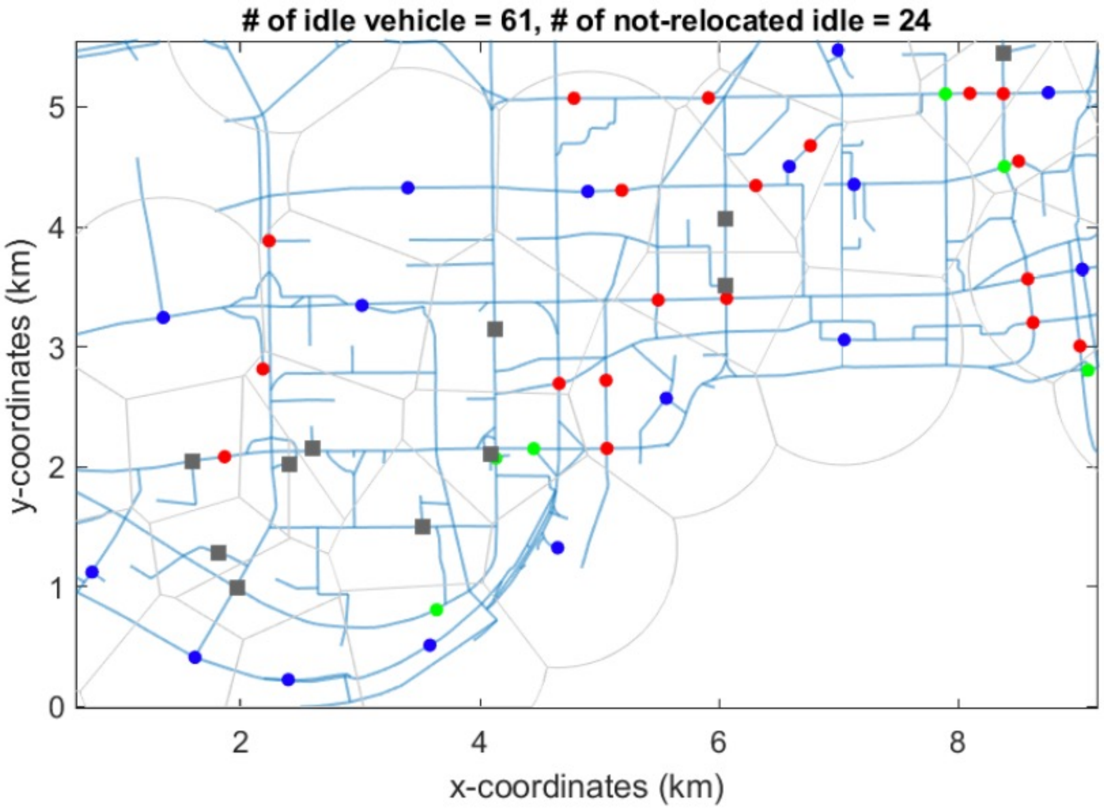
Results and Analysis



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?

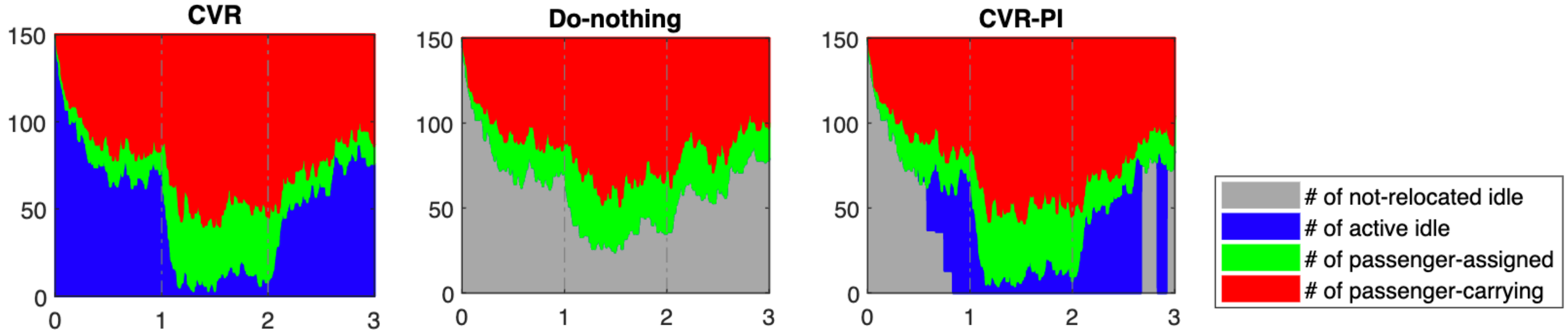
Pros

- Enhanced answer rate
- Reduced waiting times

Cons:

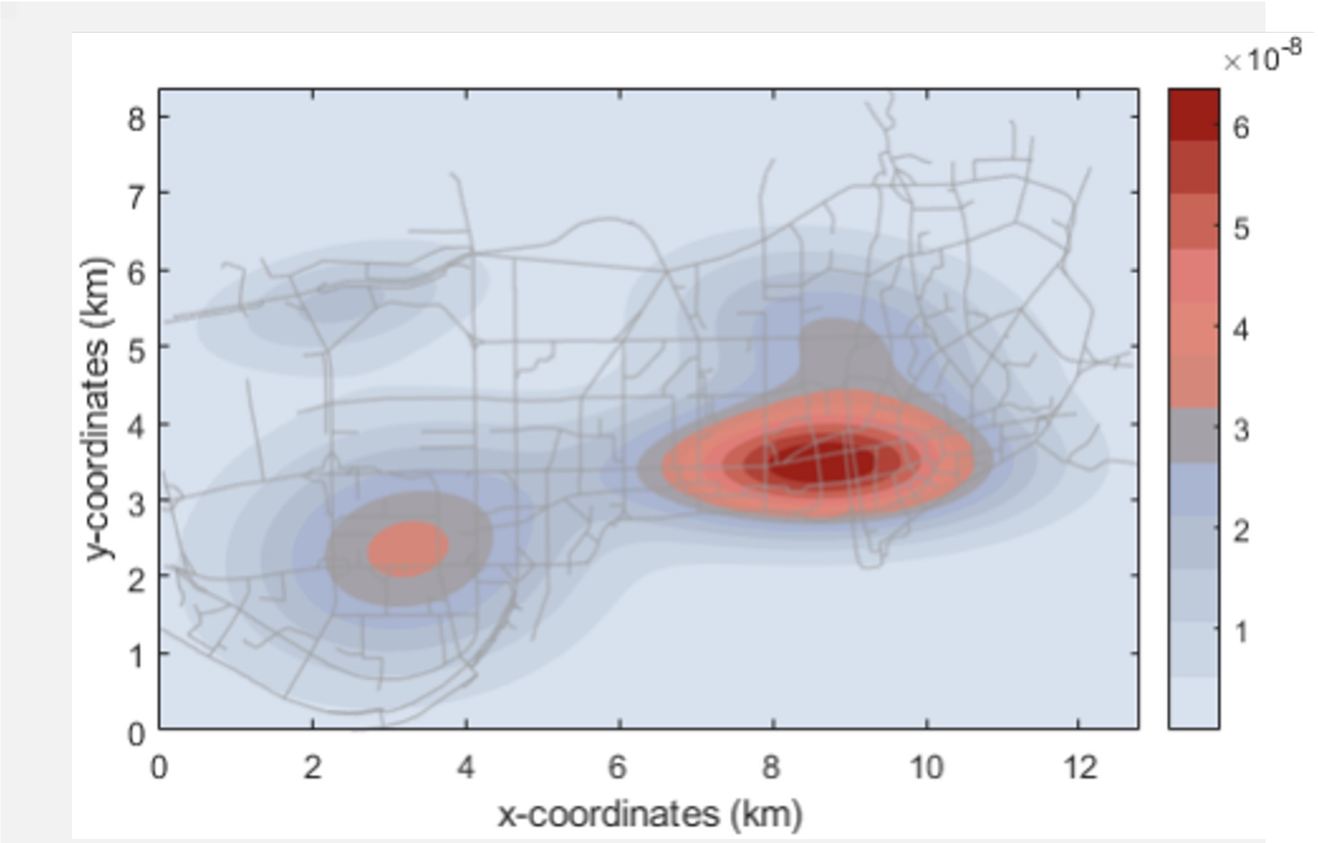
- Extra energy or fuel cost
- Congestion
- Emissions

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



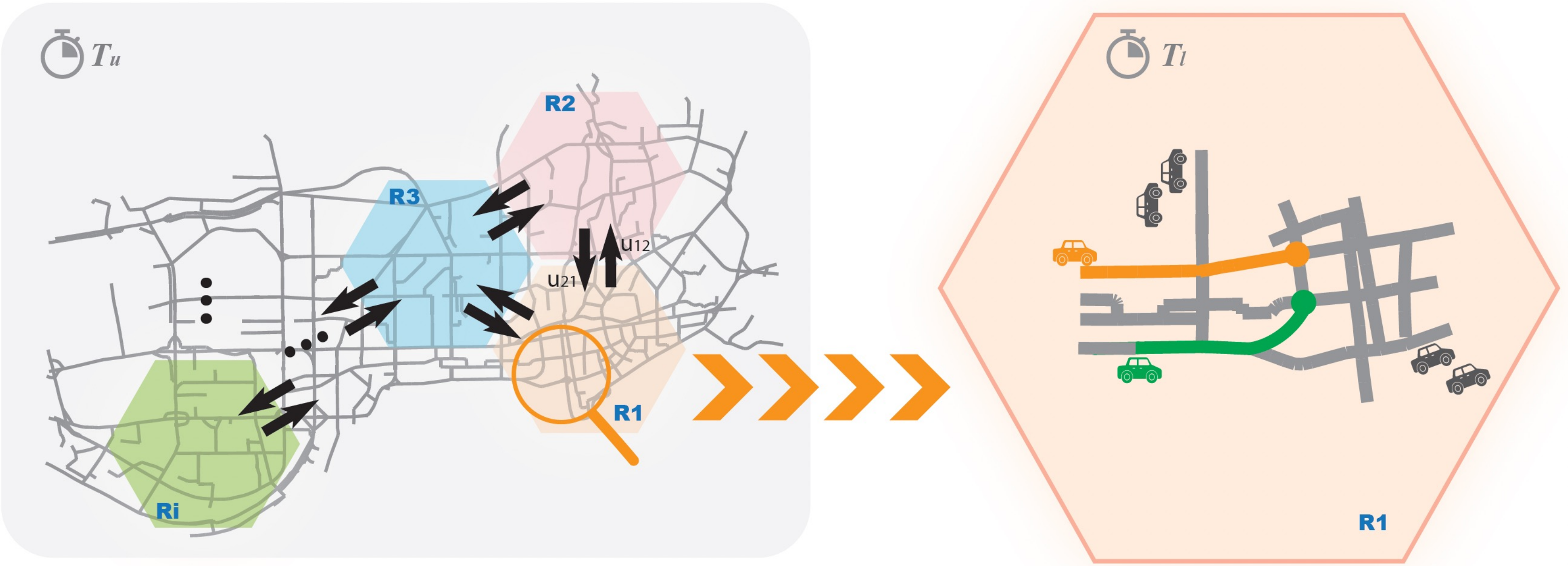
	Answer rate (%)	Average waiting time (s)	Rebalancing distance (km)
CVR	**82.6**	127.0	**4804.8**
CVR-PI	**82.4**	139.4	**2314.7**
Do-nothing	73.1	155.1	0

Drawback of use coverage control only: it may lack coordination of other regions



blue: empty vehicles

Hierarchical structure for Vehicle Rebalancing



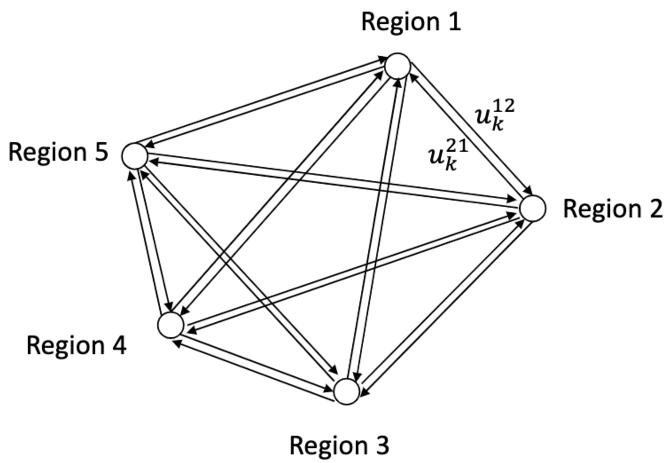
High Level (Regional Level)

Low Level (Vehicle Level)

High-level:
Control vehicle transferring between regions

Low-level:
Give precise position guidance within each region

DeePC for Vehicle Rebalancing



Schematic diagram of inter-regional rebalancing

Assume AMoD system is a LTI system with measurable disturbance:

$$\begin{cases} x_{k+1} = Ax_k + Bu_k + B_d w_k \\ y_k = Cx_k + Du_k + D_d w_k, \end{cases}$$

Control input

u_{IJ} : No. of empty vehicles be relocated from Region I to Region J.

Output

y_I : No. of successfully answered requests in Region I.

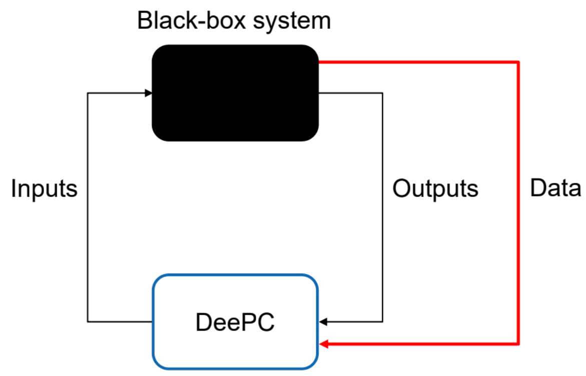
External disturbance

w_I^o : No. of requests which starts from Region I,
w_I^D : No. of requests which ends in Region I.

Review of DeePC

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

- Non-parametric method
- Use input/output trajectories
- 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-q/view

Review of DeePC

- A discrete-time linear time-invariant (LTI) system:

$$\begin{cases} x_{t+1} = Ax_t + Bu_t \\ y_t = Cx_t + Du_t \end{cases}$$

- 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:

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

u is persistently exciting of order L if $H_L(u)$ has full row rank.

Review of DeePC

- A discrete-time linear time-invariant (LTI) system:

$$\begin{cases} x_{t+1} = Ax_t + Bu_t \\ y_t = Cx_t + Du_t \end{cases}$$

- Historical input and output data of length T:

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

- Construct input/output Hankel matrix:

the first T_{ini} block row
the last N block row

$$\begin{bmatrix} U^p \\ U^f \end{bmatrix} := \mathcal{H}_{T_{ini}+N}(u^d),$$

$$\begin{bmatrix} Y^p \\ Y^f \end{bmatrix} := \mathcal{H}_{T_{ini}+N}(y^d).$$

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, sufficeiently rich previous trajectories"[2]

[1] Willems et al. (2005), "A note on persistency of excitation."
[2] Jeremy Coulson, John Lygeros, and Florian Dörfler. Sparse learning workshop 2020.

Review of DeePC

MPC:

$$\underset{u,x,y}{\text{minimize}} \sum_{k=0}^{T_f-1} \left(\left\| y_k - r_{t+k} \right\|_Q^2 + \|u_k\|_R^2 \right)$$

subject to

$$\begin{aligned}
& x_{k+1} = Ax_k + Bu_k, \ \forall k \in \{0, \dots, T_f - 1\}, \\
& y_k = Cx_k + Du_k, \ \forall k \in \{0, \dots, T_f - 1\}, \\
& x_{k+1} = Ax_k + Bu_k, \ \forall k \in \{-T_{\text{ini}}, \dots, -1\}, \\
& y_k = Cx_k + Du_k, \ \forall k \in \{-T_{\text{ini}}, \dots, -1\}, \\
& u_k \in \mathcal{U}, \ \forall k \in \{0, \dots, T_f - 1\}, \\
& y_k \in \mathcal{Y}, \ \forall k \in \{0, \dots, T_f - 1\}.
\end{aligned}$$

DeePC:

$$\underset{g,u,y}{\text{minimize}} \sum_{k=0}^{T_f-1} \left(\left\| y_k - r_{t+k} \right\|_Q^2 + \|u_k\|_R^2 \right)$$

subject to

$$\mathscr{H}_{T_{\text{ini}}+T_f} \begin{pmatrix} \hat{u} \\ \hat{y} \end{pmatrix} g = \begin{pmatrix} \hat{u}_{\text{ini}} \\ \hat{y}_{\text{ini}} \\ u \\ y \end{pmatrix},$$

$$\begin{aligned}
& u_k \in \mathcal{U}, \ \forall k \in \{0, \dots, T_f - 1\}, \\
& y_k \in \mathcal{Y}, \ \forall k \in \{0, \dots, T_f - 1\}.
\end{aligned}$$

Predictive model and state estimation in 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

1. Answer more requests

2. Consider rebalancing cost

$$= \sum_{i=0}^{N-1} -\|Qy_{k+i}\| + \|Ru_{k+i}\|$$

Economic Control Objective

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

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

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

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

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

Problem Formulation

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

Economic Control Objective

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

Regularization terms:

1. Feasibility
2. Avoid overfitting

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

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

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

Problem Formulation

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

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

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 avaliability

Low-level controller

High Level (Regional Level)

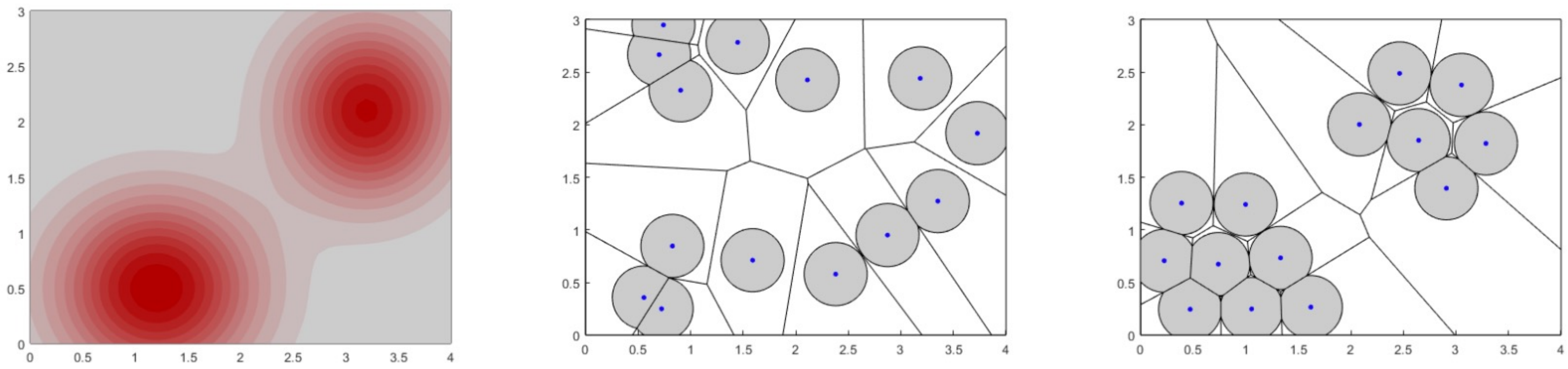
Low Level (Vehicle Level)

High-level:
Control vehicle transferring between regions

Low-level:
Give precise position guidance within each region

Low-level controller: Coverage Control-based

“What is the demand is not evenly distributed in one region?”



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

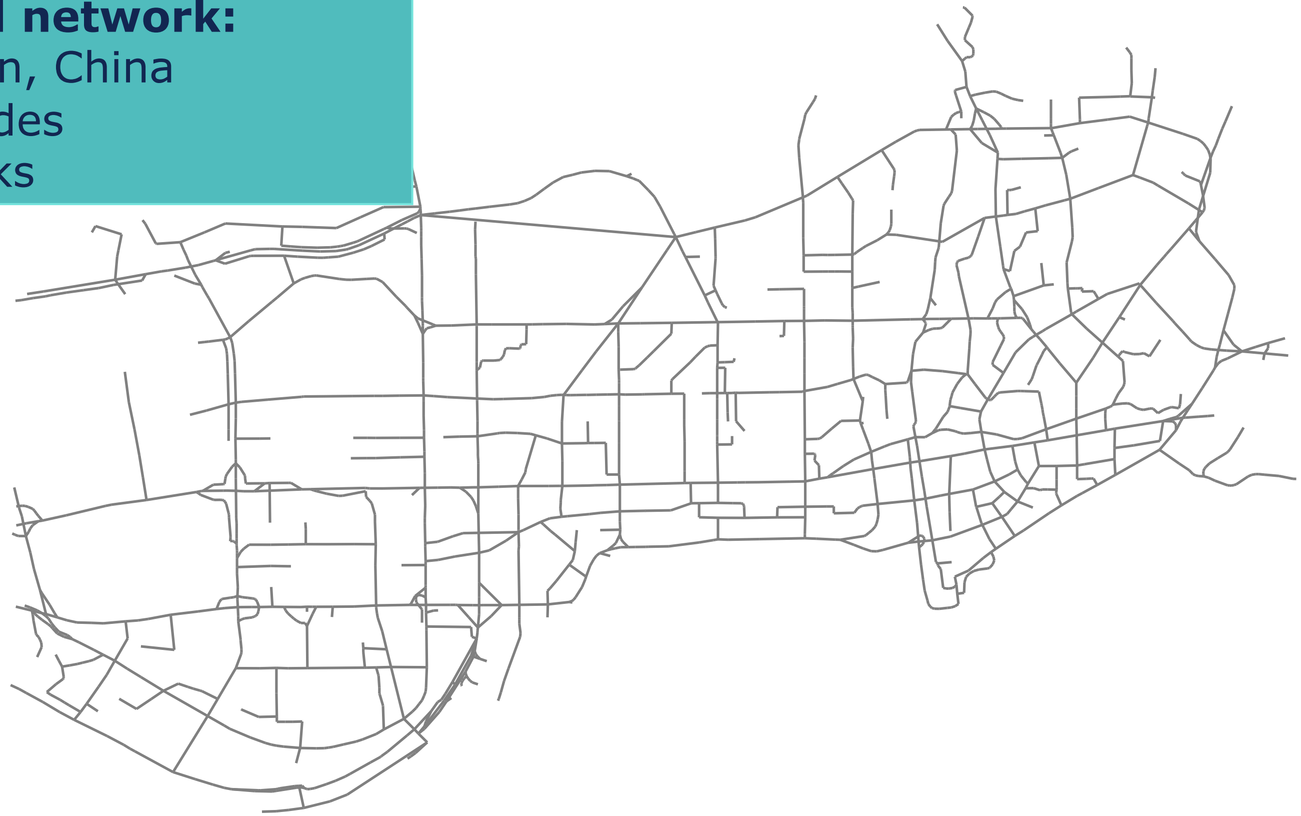
Coverage control algorithm has been applied in mobile multi-agent problems[1].

[1] Zhu et al. (2024), *"A Coverage Control-Based Idle Vehicle Rebalancing Approach for Autonomous Mobility-on-Demand Systems."*

Case Study

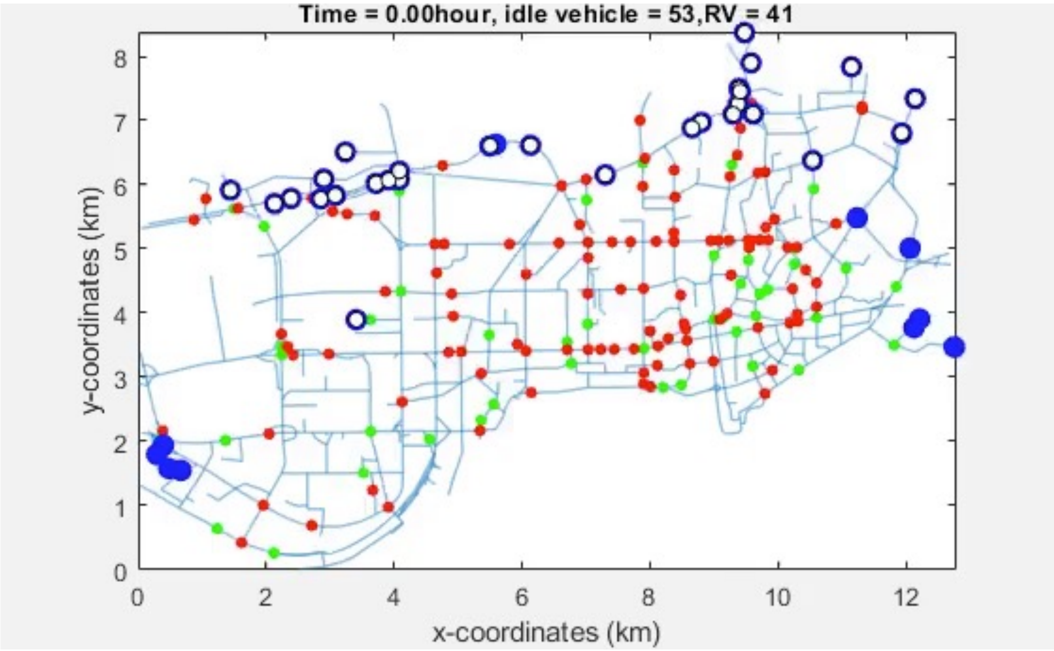
Simulated network:

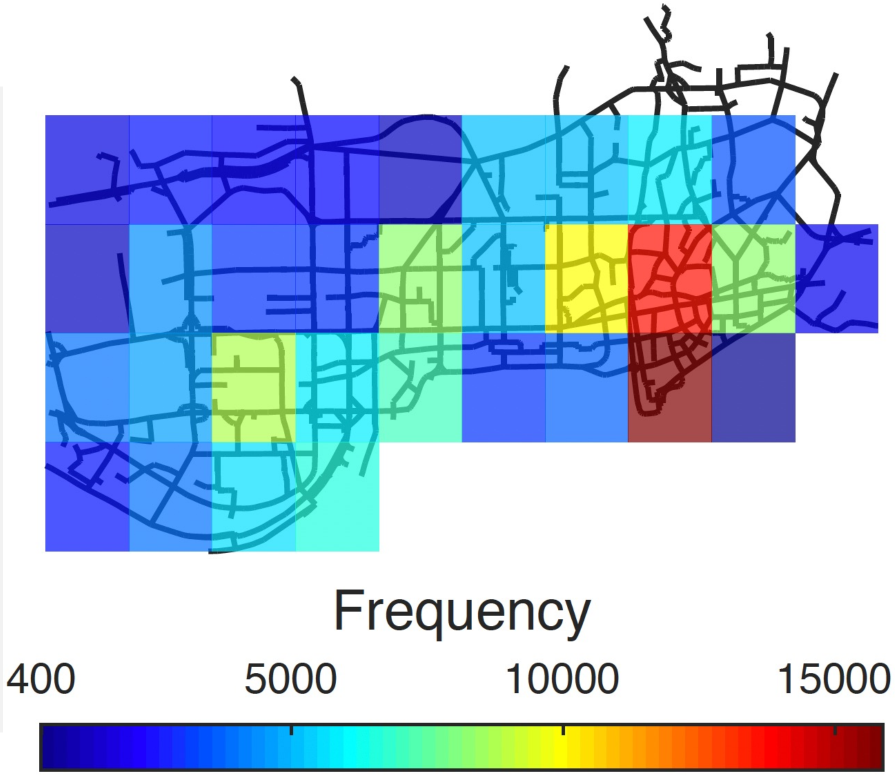
- Shenzhen, China
- 1858 nodes
- 2013 links



[1] Beojone et al.(2020), *"On the inefficiency of ride-sourcing services towards urban congestion."*

Demo video





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 and Analysis

- 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

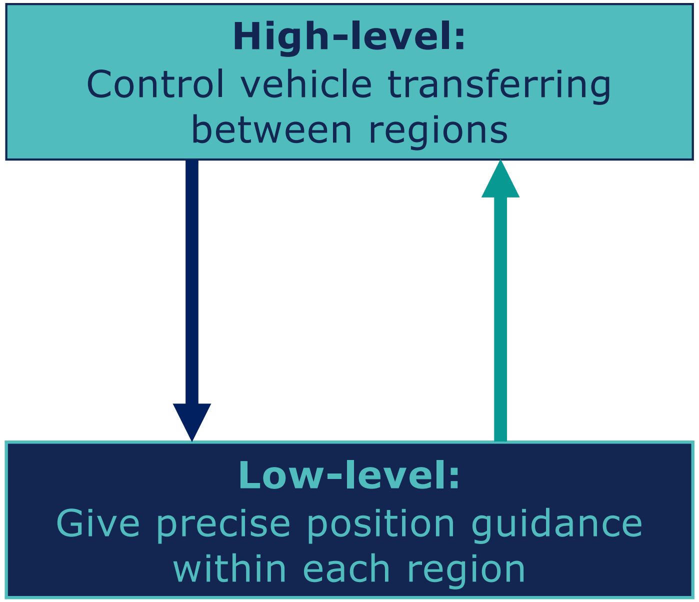
NB:

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.

Conclusions

Hierarchical control structure

bridges **macroscopic** and **microscopic** scopes.



- Advanced control methods can help with efficient fleet management, to answer more requests with less waiting time.

- 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.

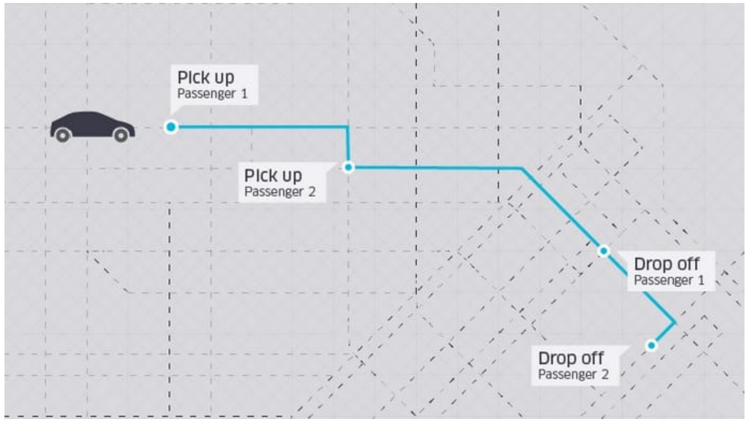
Is the shortest path the best?

— Path planning for Ride-sharing Vehicles

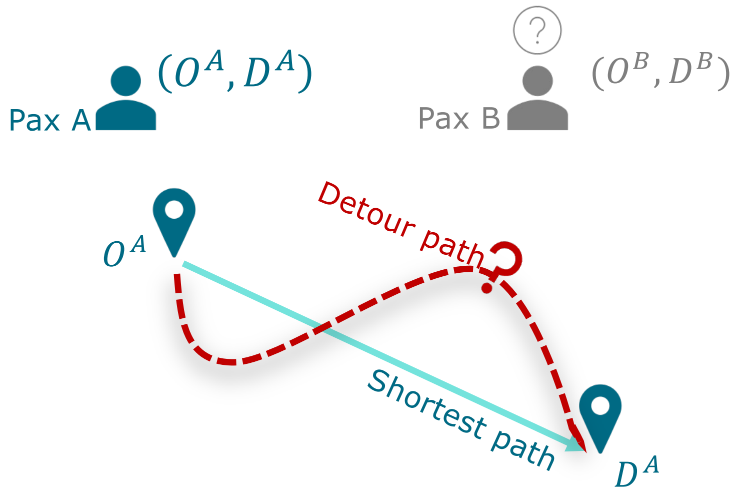
Pengbo ZHU, Giancarlo Ferrari-Trecate, Nikolas Geroliminis

EPFL

Motivation



Example of a Pool Trip



Efficient & Fast path planning:

- Given Start (Origin) and Goal (Destination) of Pax A;
- Find a path: higher possibility to pick up a second passenger en route, at the same time, coordinates with other vehicles.
- Satisfy constraints, such as max. delay of Pax A and Pax B;

What is path planning

Path Planning

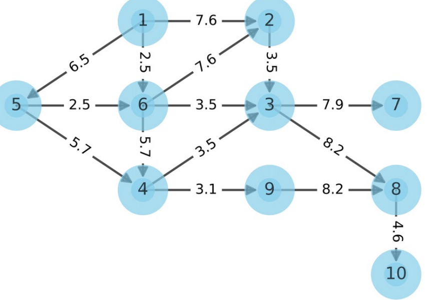
Path planning addresses the problem of finding a good path from the starting point to the goal.

—avoiding obstacles, avoiding enemies, and **minimizing costs** (fuel, time, distance, money, etc.)

Our objective:

1) Geometrical Cost

- Travelling distance/time of the planned route.



For graph $G = (Q, E, w)$,
Edge weight w is defined as the length of the road segment.

2) Likelihood of getting a second passenger

- (Attractiveness) Passenger demand distribution
- (Repulsiveness) Other vehicles: Idle & not-fully occupied vehicles

Problem Formulation

Decision Variable

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

Objective Function

The objective is to minimize:

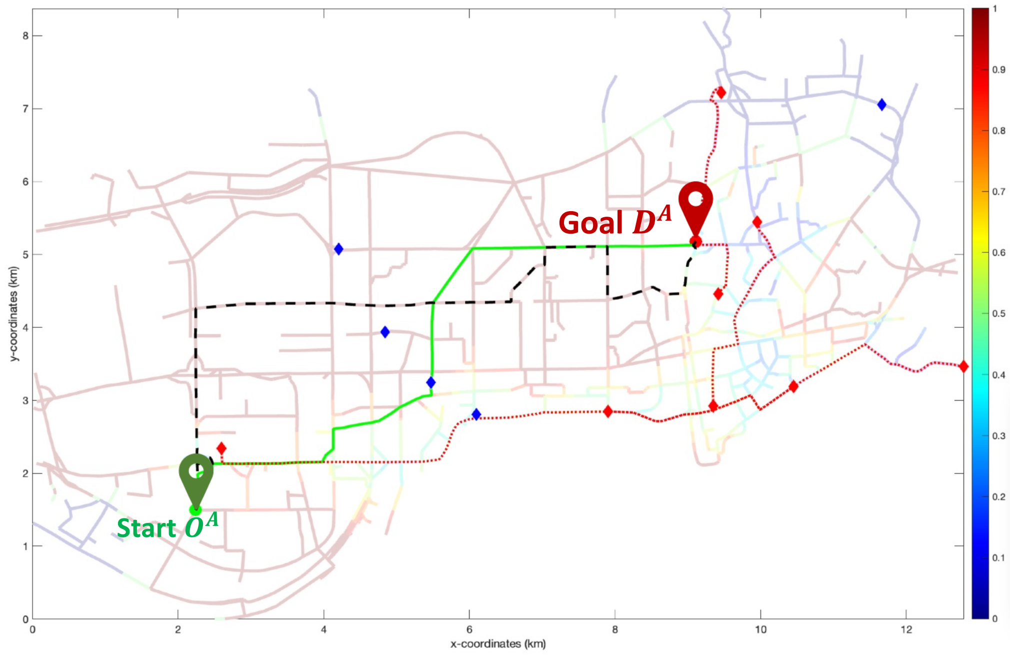
Likelihood of get a second passenger

$$J = \underbrace{\sum_{(i,j)\in E} T_{ij} x_{ij}}_{\text{Total planned travel time}} - \gamma \sum_{(i,j)\in E} \left(1 - \left(1 - p_{ij}\right)^{T_{ij}}\right) x_{ij}$$

where

- $\gamma > 0$ is a parameter to balance the trade-off between minimizing the trip travelling time and maximizing the likelihood of picking up a second passenger.
- w_{ij} is the link length associated with edge (i, j),
- $T_{ij} = \frac{w_{ij}}{\hat{v}}$ is the travel time on edge (i, j),
- p_{ij} is the adjusted probability of getting a second passenger on edge (i, j) for a time unit.

Case Study



Proposed: 11.91 km

Shortest (Travel time Only): 9.54km

Our proposed method designs a path that:

1. Avoid ‘overlapping’ with other planned routes,
2. Keep distance from other empty vehicles,
3. Pass through higher demand area,
4. An acceptable detour distance.

Blue diamonds: Idle vehicles

Red diamonds: Not fully-occupied vehicles

Red dotted lines: planned routes of not fully-occupied vehicle

Case Study

Fleet size = 200; all order: 7200

	Answer rate (%)	Waiting time (s)	Shared trip	Av. Extra trav time (s)
Proposed	82.2	118.1	3181	301.3
No share	66.9	189.5	NaN	0

Take home message:

- Design a path planning algorithm for not-fully-occupied vehicles
  1) Consider the passenger demand (Attractiveness),
  2) Coordinate with other vehicles (Repulsiveness).
- Computation load is light.
- Better service with higher answer rate and shorter waiting time.