IntelliLight: A Reinforcement Learning Approach for Intelligent Traffic Light Control

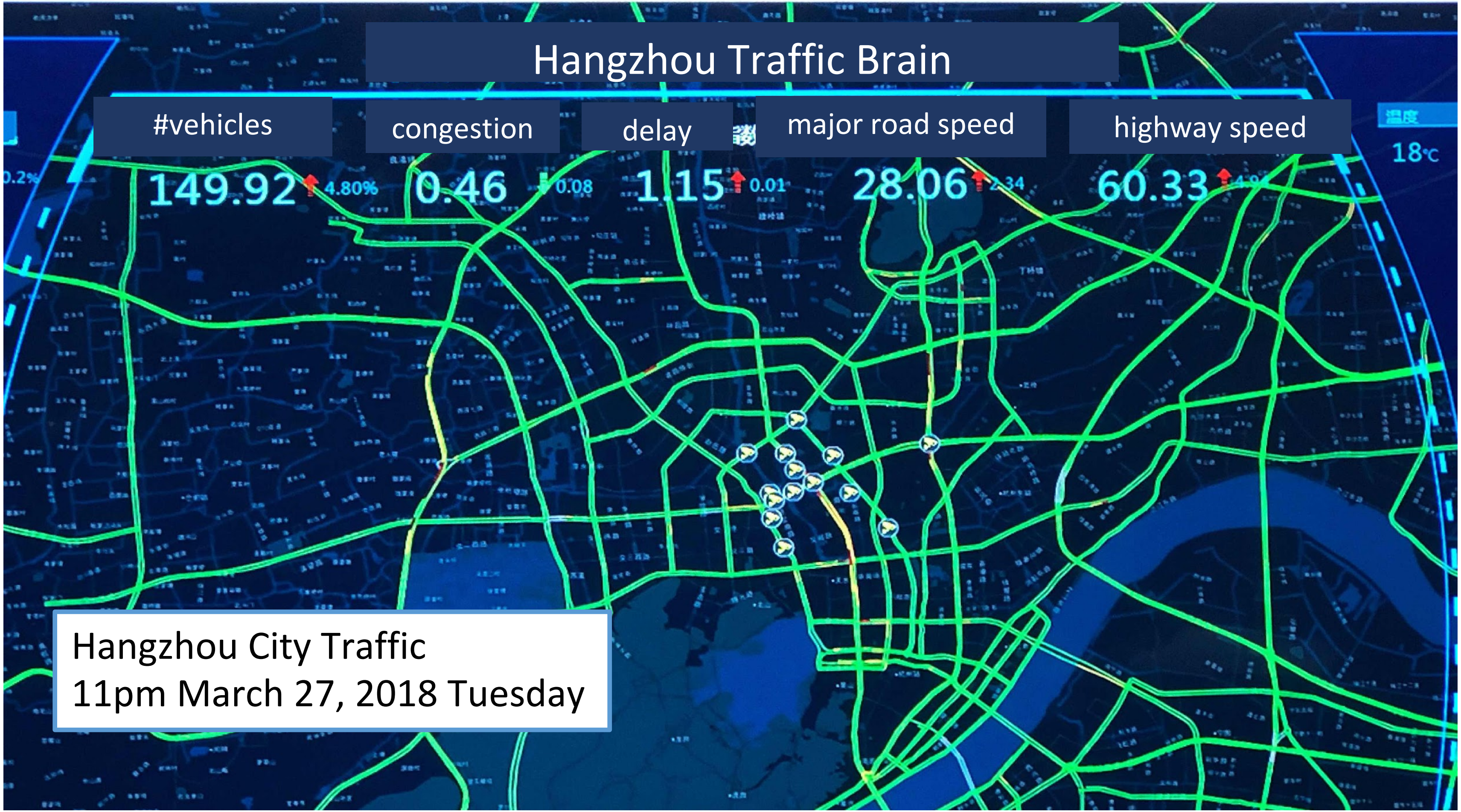
Hua Wei*

Guanjie Zheng*

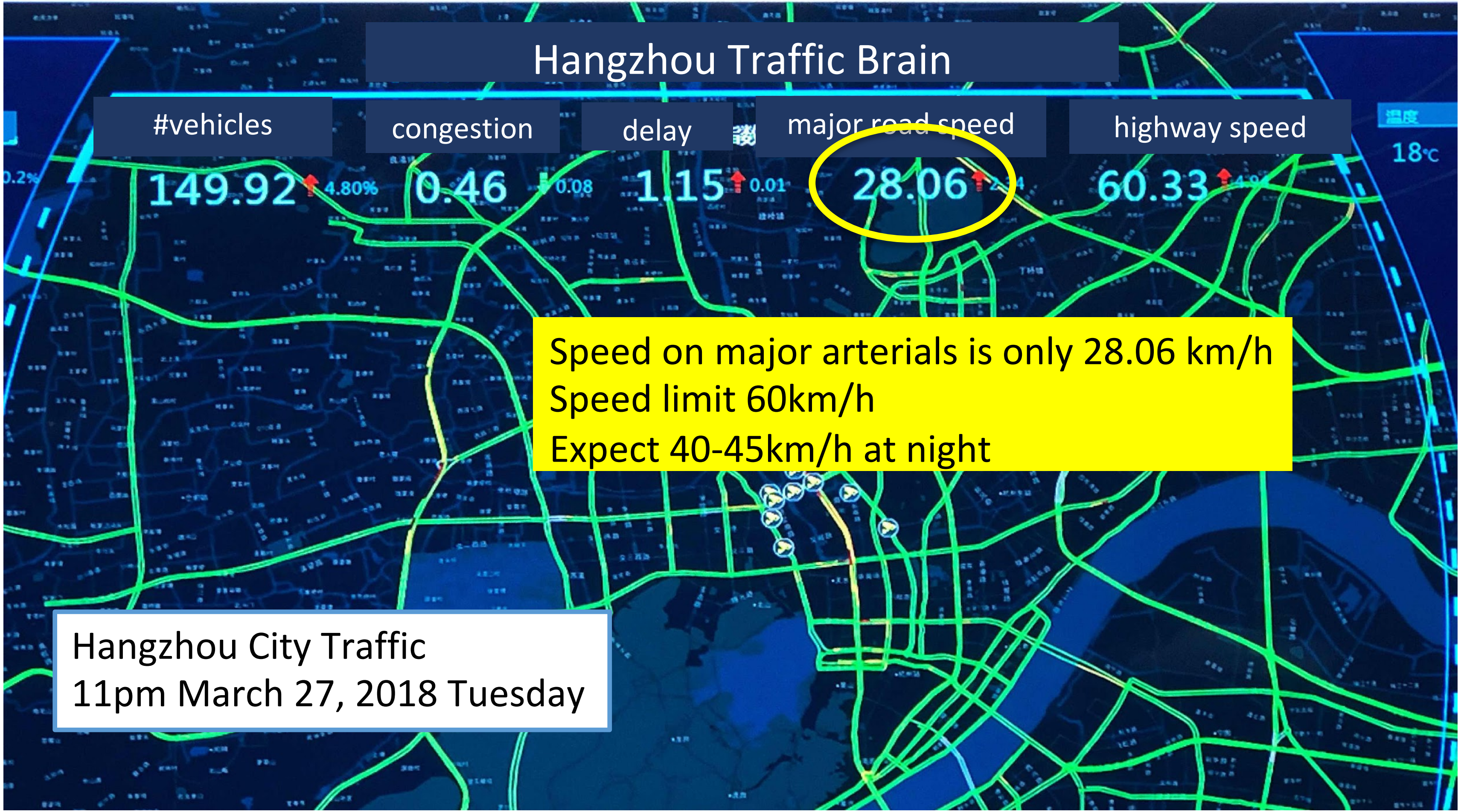
Huaxiu Yao

Zhenhui (Jessie) Li

* Equal contribution


Hangzhou Traffic Brain
#vehicles
congestion
delay
major road speed
highway speed
149.92
4.80%
0.46
0.08
1.15
0.01
28.06
60.33
18℃
Hangzhou City Traffic
11pm March 27, 2018 Tuesday

Hangzhou Traffic Brain



#vehicles	congestion	delay	major road speed	highway speed

Speed on major arterials is only 28.06 km/h
Speed limit 60km/h
Expect 40-45km/h at night

Hangzhou City Traffic
11pm March 27, 2018 Tuesday

Traffic light fails to see the traffic

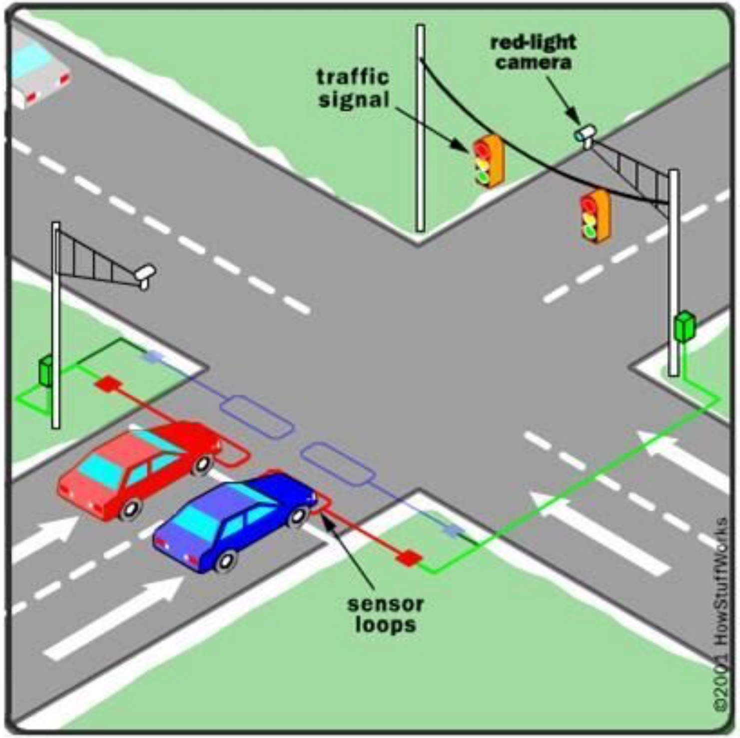
Traffic light fails to see the traffic

Spillover

Light through movement traffic
heavy left-turn traffic

The current traffic signal control systems

- SCATS
  - Sydney Co-ordinated Adaptive Traffic System
  - Developed in 1980s
  - Each traffic signal: 8~16 manually designed signal plans, not learned by data
  - Use loop sensor data to choose the plan



Why today? (we could improve traffic signal)

New rich data

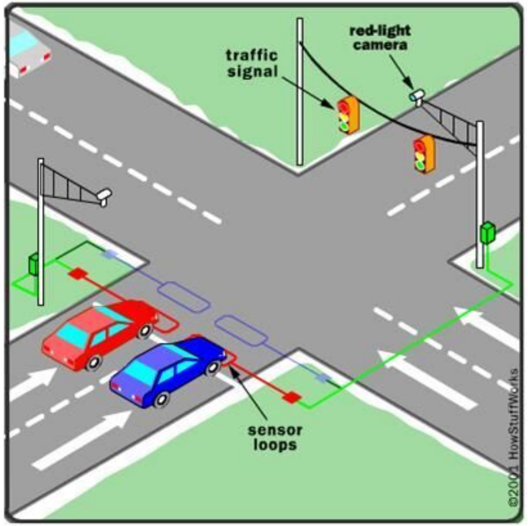
- Loop sensor data vs. camera data

New data-driven models

- Transportation models vs. machine learning data-driven models

Why today? New rich data

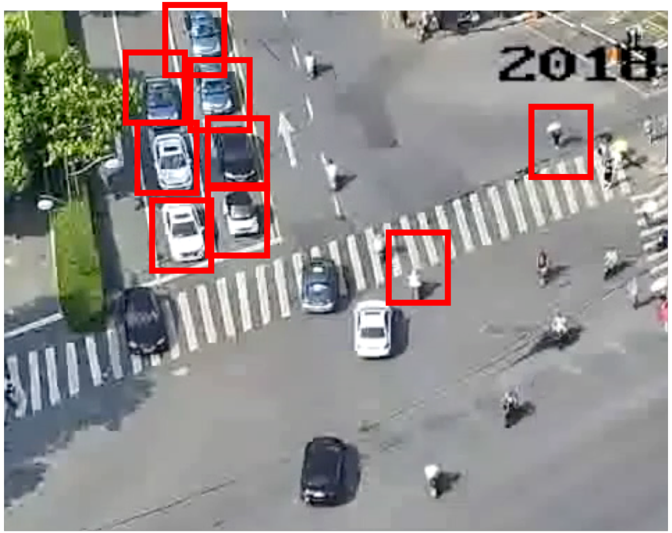
Yesterday



Loop sensor data

Only count the vehicle when it passes the sensor

Today



Camera data

Show positions of all vehicles, pedestrian, and bicycles

Why today? New data-driven model

Traditional Transportation

Optimization under assumptions of traffic model

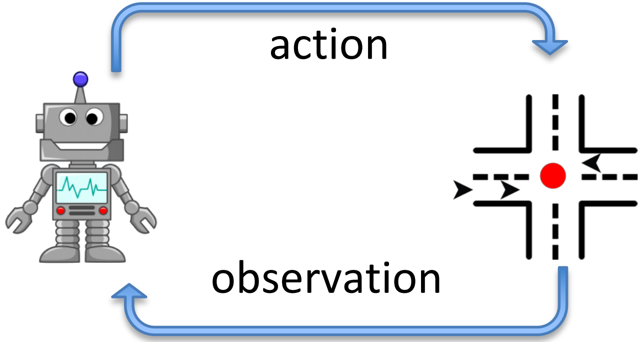
minimize *duration*
subject to ***traffic model***

Signal plan = func (*traffic data*)

Assumptions do not apply in the real world!

Reinforcement Learning

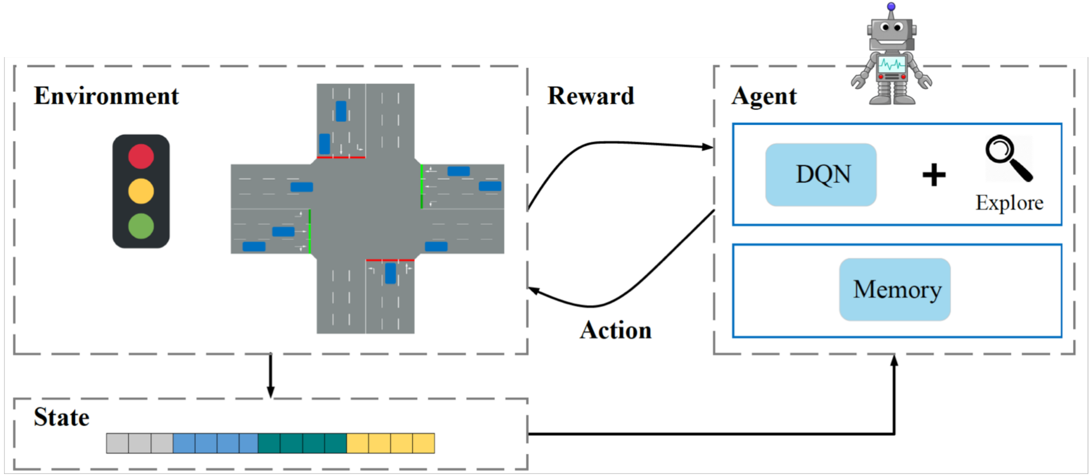
Directly learning from real-world data



Literature review in RL for signal control

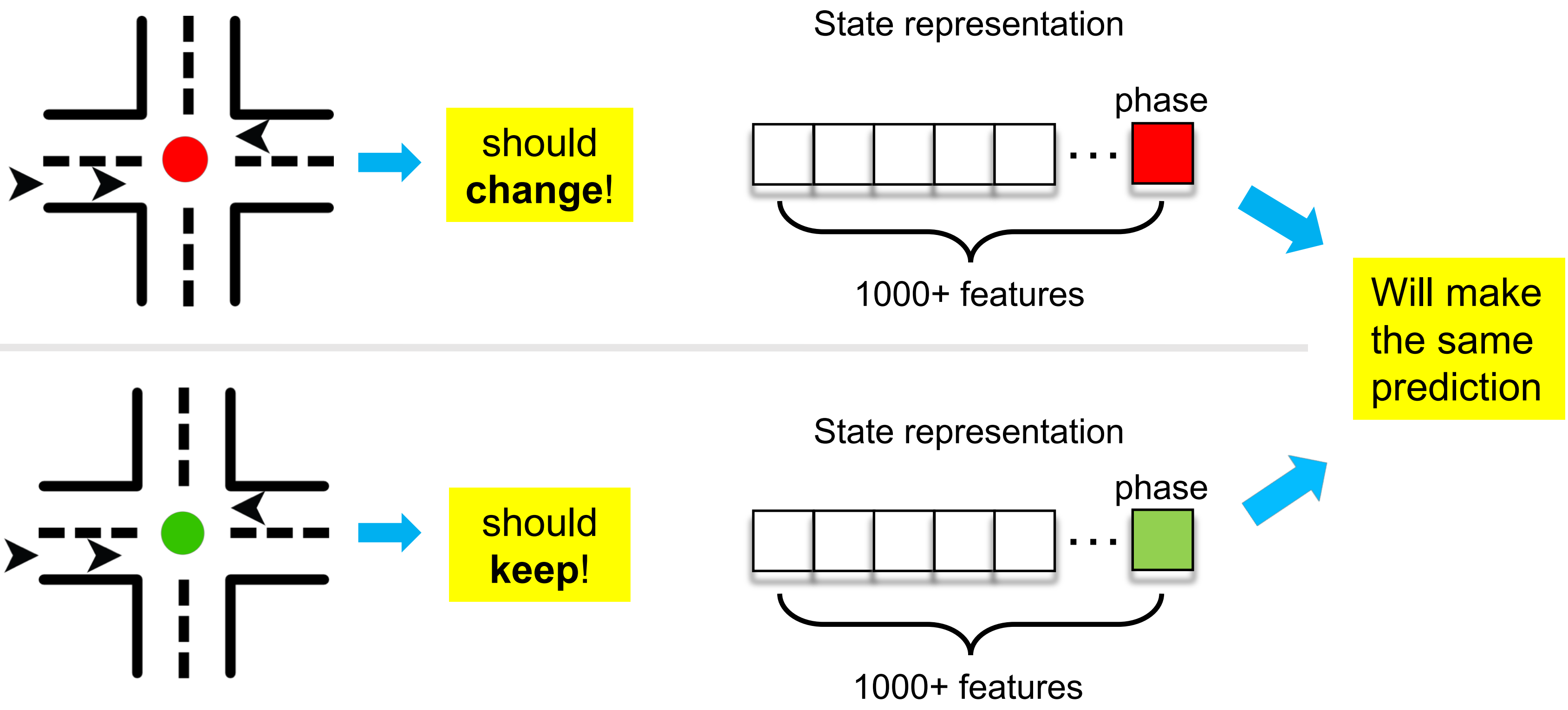
- **Tabular method (discrete state):**
  - Q-learning (El-Tantawy et. al. 2010, El-Tantawy et. al. 2012)
  - Discrete state, can not scale up
- **Approximation method (continuous state):**
  - DQN (Gao et. al. 2017, van der Pol and A. Oliehoek 2016)

Our proposed RL framework

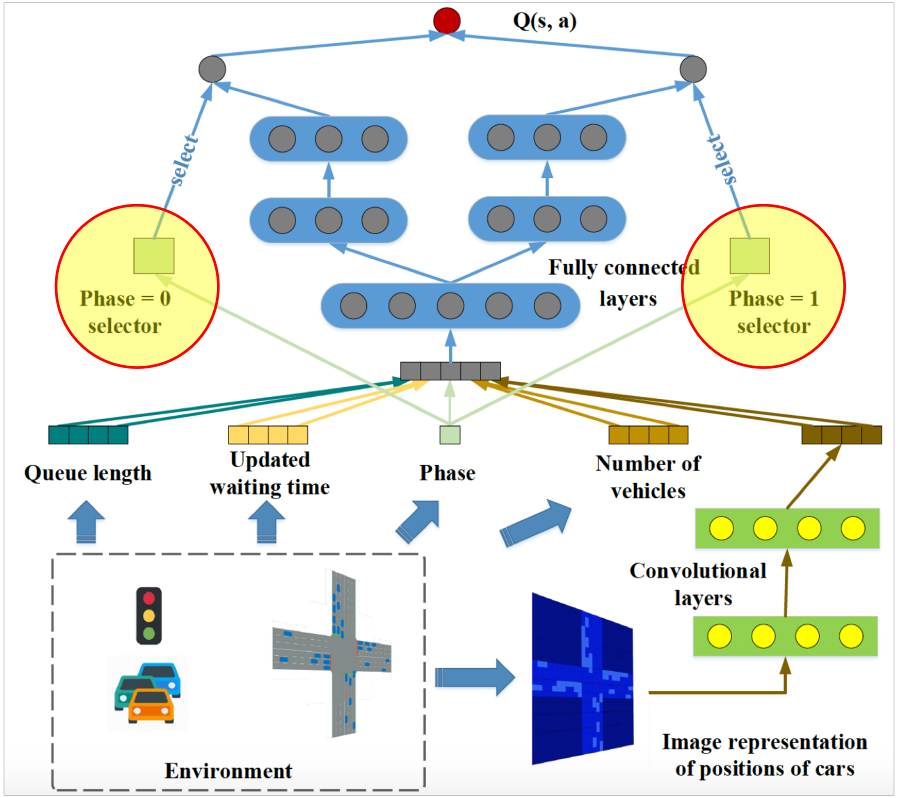


- **Setting**: One intersection, no turns
- **Reward**: queue length, average waiting time, sum of delay
- **Action**: keep the signal or change the signal
- **State**: queue length, #cars, waiting time, traffic situations (image), signal

Q1: Represent state as plain features?

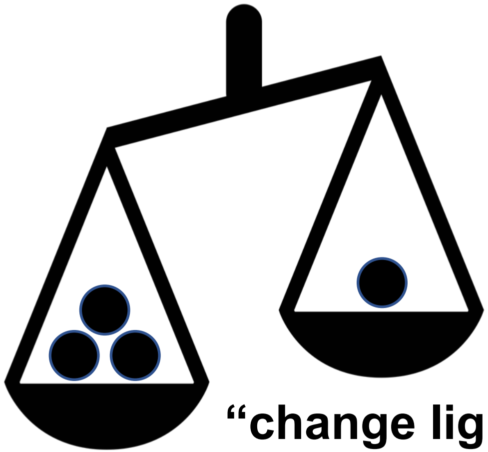


Q1-Solution: Phase-gated Deep Q-Network



Phase as a gate to separate decision making

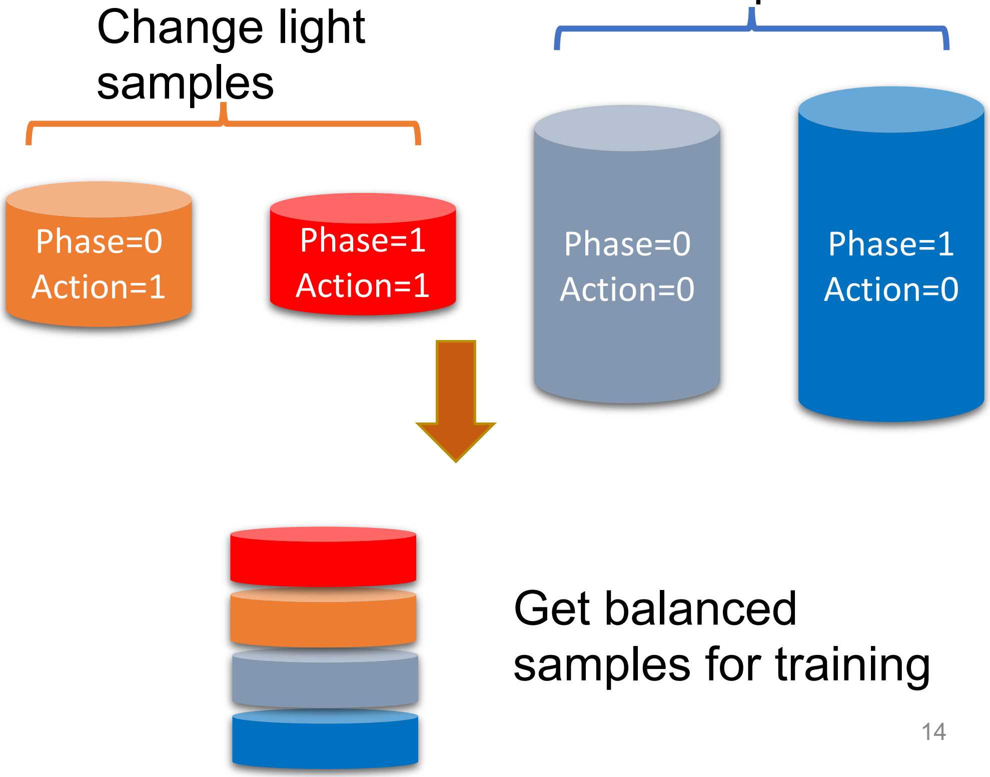
Q2: How to avoid unbalanced samples?



“change light”

“keep light”

#keep is way more than #change



Get balanced samples for training

Do these two special designs help?

Model name	Reward	Queue length	Delay	Duration
Fixedtime	-1.670	4.601	2.883	39.707
Base	-5.030	5.880	3.432	39.021
Base + MP	-3.329	5.358	2.238	44.703
Base + MP + PS	**-0.474**	**0.548**	**2.202**	**25.977**

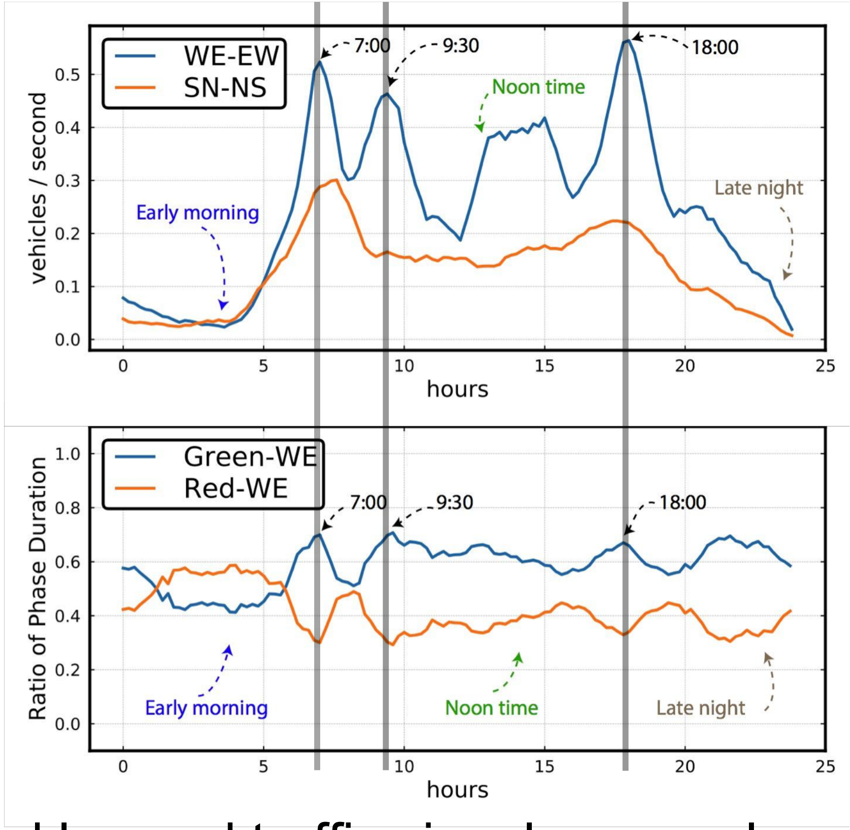
MP: memory palace **PS**: phase selector

Experiment results using real data (from a city in China)

Methods	Reward	Queue Length	Delay	Duration
FT	-5.727 ± 5.977	19.542 ± 22.405	3.377 ± 1.057	84.513 ± 60.888
SOTL	-35.338 ± 65.108	16.603 ± 17.718	4.070 ± 0.420	64.833 ± 23.136
DRL	-30.577 ± 26.242	54.148 ± 43.420	4.209 ± 1.023	166.861 ± 93.985
IntelliLight	**-3.892** ± 7.609	**10.238** ± 20.949	**2.730** ±1.086	**50.487** ± 46.439

- **FT**: Fixed Time
- **SOTL**: Self-Organizing Traffic Light Control (changing the light when #cars waiting > threshold)
- **DRL**: Deep Reinforcement Learning (van der Pol et al, 2016)

Policy learnt from real data



Traffic volume and learned traffic signal on a real world intersection

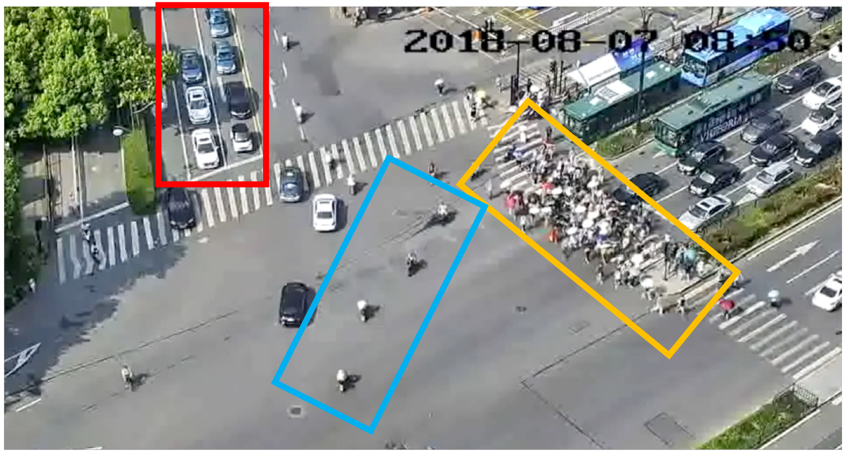
Modern city traffic is complex. We still have open questions

- How to mitigate trial-and-error cost when applying RL in real world?

https://www.swlexledger.com/single-post/2018/09/10/All-state-office-and-schools-in-Lexington-County-closed-tomorrow

Modern city traffic is complex. We still have open questions

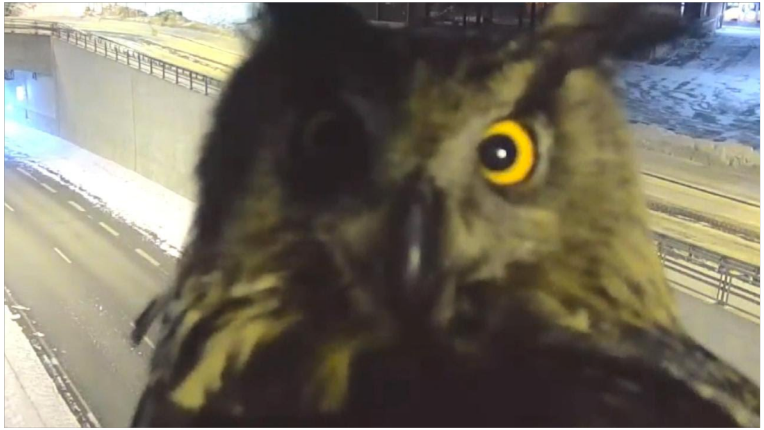
- How to mitigate trial-and-error cost when applying RL in real world?
- How to design a "fair" reward function?



Cars scooters pedestrians

Modern city traffic is complex. We still have open questions

- How to mitigate trial-and-error cost when applying RL in real world?
- How to design a “fair” reward function?
- Real data are messy and incomplete.



https://www.insideedition.com/inquisitive-owl-videobombs-traffic-camera-finland-40999

IntelliLight: A Reinforcement Learning Approach for Intelligent Traffic Light Control

Interested in working on traffic problem in Hangzhou?

https://faculty.ist.psu.edu/jessieli/

jessieli@ist.psu.edu

Thanks for your attention! Any questions?