Real-Time Intelligent Autonomous Intersection Management Using Reinforcement Learning

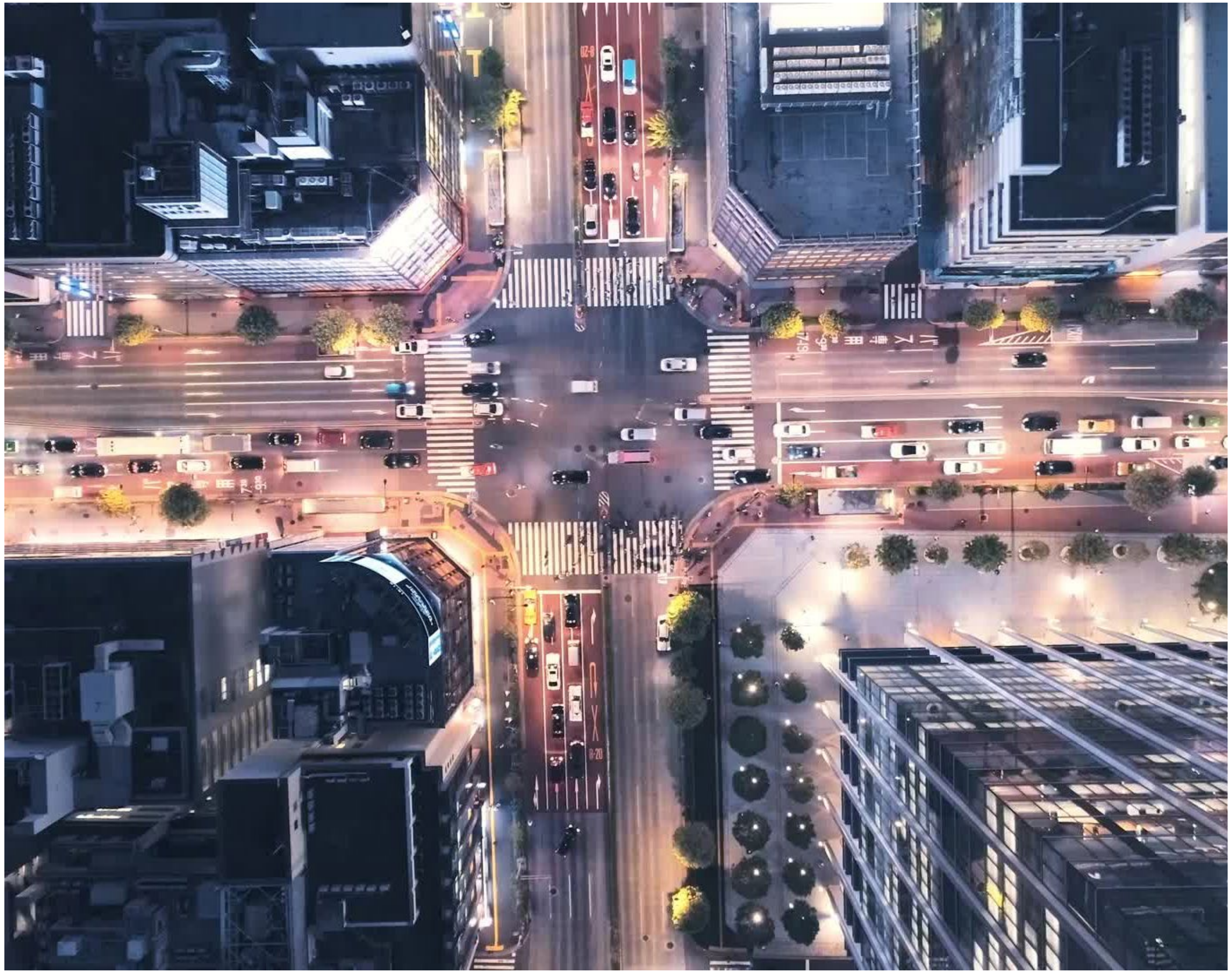
Authors: Udesh Gunarathna, Shanika Karunasekera,
Renata Borovica-Gajic, Egemen Tanin

Presenter: Udesh Gunarathna

Most of us stuck at intersections everyday!

Why?

- Stop-and-go nature at red lights disrupts the traffic flow
- There is a time gap between traffic signals phases
- We wait at red lights even when there are no vehicles crossing the intersection from other roads



Can we do better?

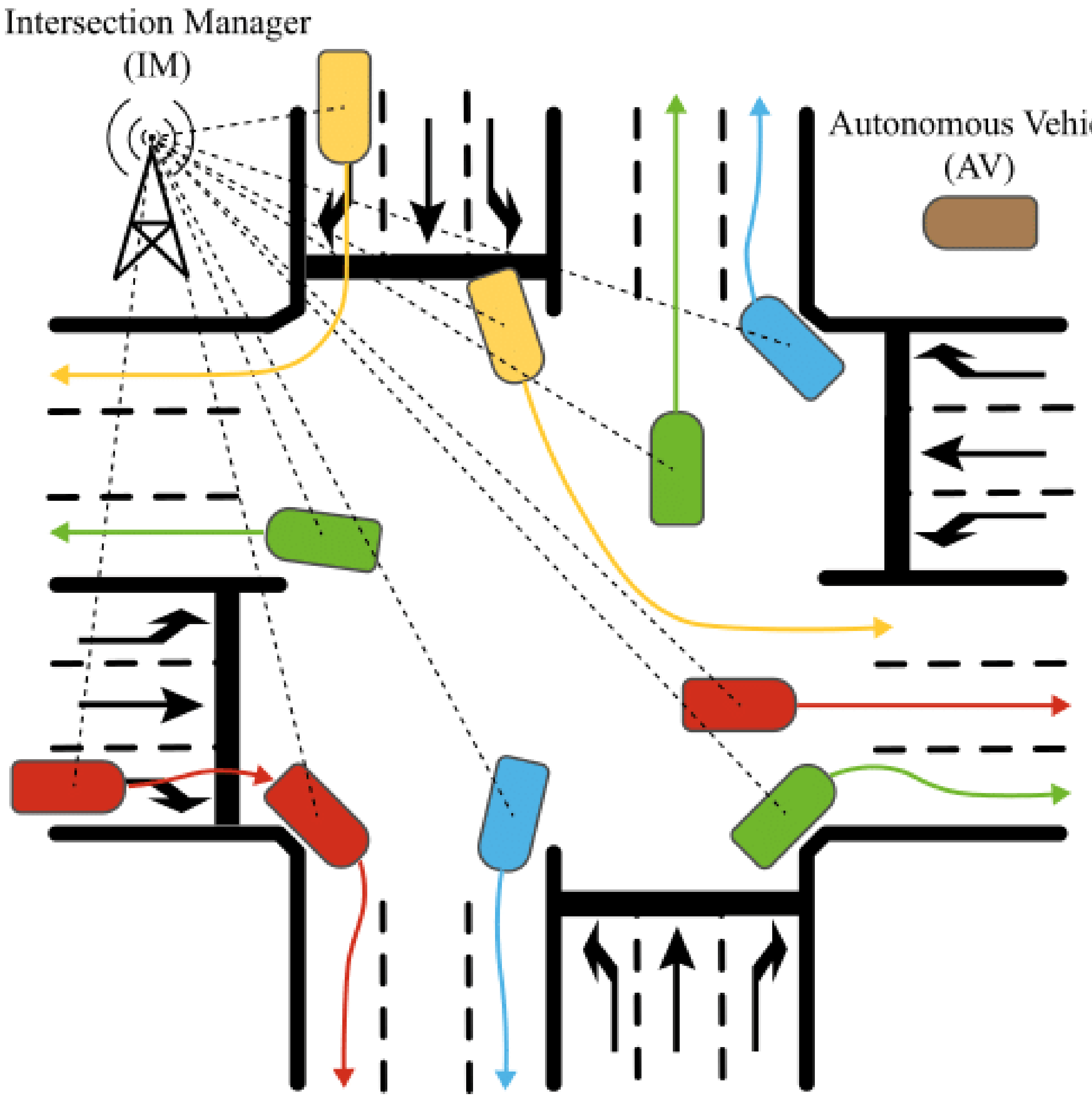
- Can we avoid traffic signals and manage to cross the intersection collaboratively?
- Challenging to coordinate with human drivers
- Autonomous vehicles and smart infrastructure can provide a solution

Autonomous Intersection Management (AIM)

- Vehicles coordinate their speed and arrival times towards the intersection for collision free traversal through the intersection

- Vehicles traverse in First-Come First-Server order

- Proposed by Dresner *et al.*[1]



www.mdpi.com/1424-8220/22/6/2217

[1]. K. Dresner and P. Stone, "A multiagent approach to autonomous intersection management," *Journal of Artificial Intelligence Research*, vol. 31, pp. 591–656, 2008

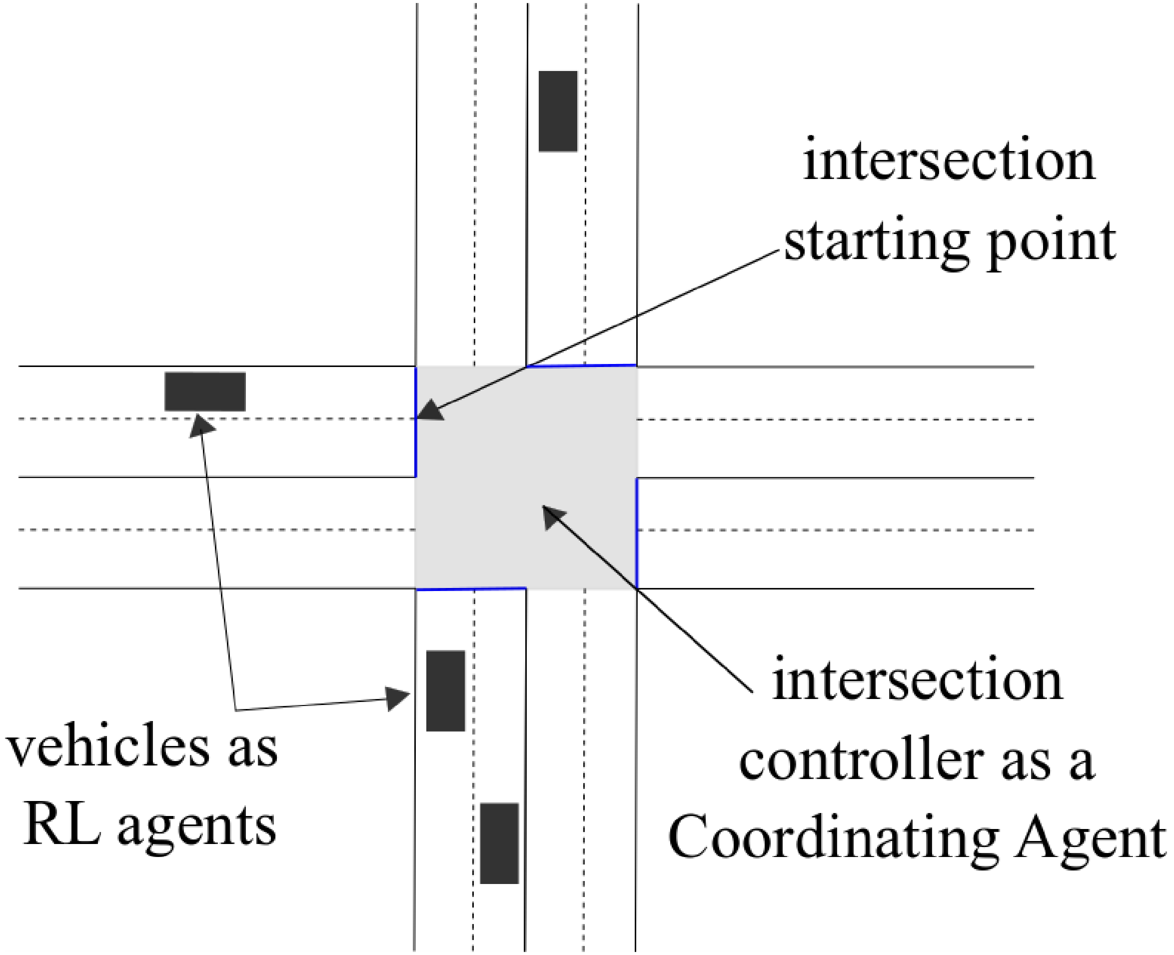
Improvements to AIM

- We can optimize the traversing order to improve the flow. i.e. leverage platooning
- If vehicles cross the intersection at a higher speed; the intersection crossing time is reduced and stop-and-go nature is avoided

Challenges of AIM in Real-time

- Computational time should be low
- If the intersection controller takes a long time to compute a schedule which may lead to vehicle crashing
- Autonomous vehicles need to plan their trajectories in real-time

Proposed Architecture: A Multi-agent Solution



- Models the problem as a multi-agent solution to overcome the computational complexity

- Coordinating Agent uses a modified polling-based system to find an optimized order to travel and assign a schedule for each vehicle

- RL Agents use a novel reinforcement learning algorithm to compute the best trajectory to adhere to the schedule

Polling-based Coordinating Agent

Objective: Find an optimized traversing order for vehicles arriving at the intersection stochastically and assign an arrival time for each vehicle

Models the vehicles as a set of customers in a set of queues and the intersection controller as a server

The server computes the service order by popping customers from queues. It spends a **service time** for each customer and encounters **switching time** when a customer is selected from a different queue

Modified Polling System

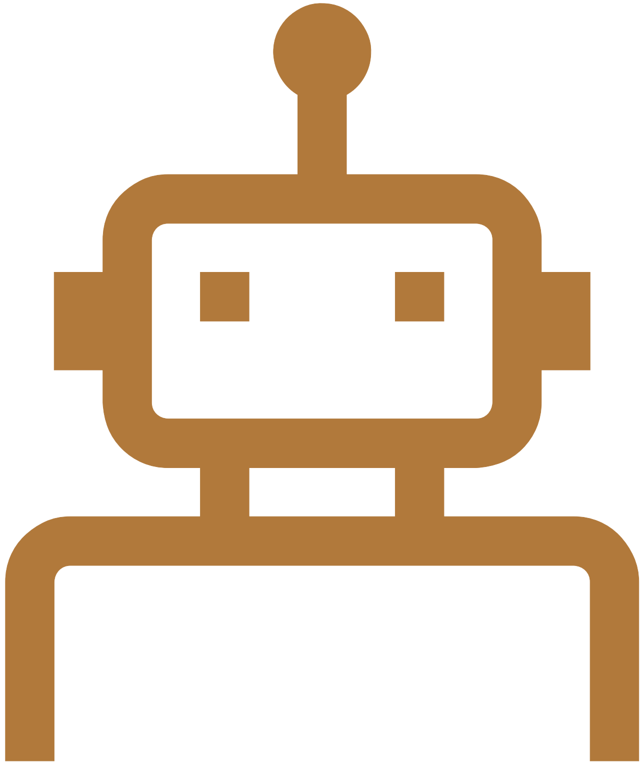
- The above formulation can only be applied to intersections with single lanes and vehicles going straight through the intersection
- We proposed a modified polling system to handle complex intersections with multiple-lanes and multiple turning directions
- Our polling algorithm proposed a **queue dependent matrix** to replace switching time and service time

Reinforcement Learning Agents

- **Objective 1 (Trajectory Control Task):** Compute a trajectory to reach the intersection exactly at the scheduled time
  - If a vehicle reaches in a higher speed to the intersection, then it will take less time to cross the intersection

- **Objective 2 (Cruise Control Task):** Needs to be aware of the vehicle in front and maintain a safe distance

Why achieving both tasks is difficult



- **Trajectory control task** requires long term planning while **Cruise control task** requires short term planning

- Typically, reinforcement learning algorithms learn either the short-term or long-term goal

- Q-learning contains a fixed parameter named discount factor to set the problem as a short-term or long-term task but cannot accommodate both simultaneously

Solution?

- We propose a novel RL algorithm named **Multi-discount Q-learning** to solve the above problem without increasing the computational complexity to learn the task

- Multi-discount Q-learning adaptively changes discount factor based on the feedback received from the trajectory control and cruise control tasks

- Can guarantee the convergence

Experimental Results: Performance

Traffic Level	Low	Mid	High
DTS	77.59	235.73	543.18
FCFS-AIM	15.21	131.46	388.57
H-AIM	15.25	98.09	313.34
LP-AIM	13.85	94.92	304.44
CMQ-AIM	**13.66**	**92.76**	**302.63**

Experiments were carried out in a four-legged intersection. The travel time is shown in seconds

Experimental Results: Safety

Traffic Level	Low	Mid	High
H-AIM	65	49	69
LP-AIM	1	66	51
CMQ-AIM	0	0	0

Experiments were carried out in a four-legged intersection.
The number of vehicles violated the arrival time are presented in the table

Detailed results available in the paper!

- **Thank You!**