ACTORRL: A NOVEL DISTRIBUTED REINFORCEMENT LEARNING FOR AUTONOMOUS INTERSECTION MANAGEMENT

Guanzhou Li
Tsinghua University
Beijing, China
ligz19@mails.tsinghua.edu.cn

Jianping Wu*
Tsinghua University
Beijing, China
jianpingwu@tsinghua.edu.cn

Yujing He
Tsinghua University
Beijing, China
hyj19@mails.tsinghua.edu.cn

ABSTRACT
As an emerging tendency of future transportation, Connected Autonomous Vehicle (CAV) has the potential to improve traffic capacity and safety at intersections. In autonomous intersection management (AIM), distributed scheduling algorithm formulates the interactions among traffic participants as multi-agent problem with information exchange and behavioral cooperation. Deep Reinforcement Learning (DRL), as an approach obtaining satisfying performance in many domains, has been brought in AIM recently. Attempts to overcome the challenges of curse of dimensionality and instability in multi-agent DRL, we propose a novel DRL framework for AIM problem, ActorRL, where actor allocation mechanism attaches multiple roles with different personalities to CAVs under global observation, including radical actor, conservative actor, safety-first actor, etc. The actor shares behavioral policies with collective memories from CAVs it is assigned to, playing the role of “navigator” at AIM. In experiments, we compares the proposed method with several widely used scheduling methods and distributed DRL without actor allocation, the results shows better performance over benchmarks.

Keywords ActorRL · Autonomous intersection management(AIM) · Multi-agent RL · Distributed RL · Connected Autonomous Vehicle (CAV)

1 Introduction
In the urban traffic circumstance, intersections are major locations for traffic delays and accidents[1]. Interweaving of traffic flows from different directions and complex interactions of various traffic participants make intersection an accident-prone scenario. Statistics from U.S. Federal Highway Administration declare 40 percent of crashes were related to intersections. The advent of Connected Autonomous Vehicle (CAV), building upon Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) communication technologies, provides a new perspective for the safe and efficient passage at intersections. In terms of the communication protocols and schedule algorithms, the existing intersection management can be categorized into centralized and distributed methods. The former obtains vehicle’s status mainly through V2I protocol and the intersection manager (IM) determines the crossing order of CAVs and HVs globally. In this framework, the interactive mode between IM and CAVs can be further divided into request mode and assign mode[2]. In the request mode, CAVs actively report their expected speed or crossing time to IM then IM collects all requests to schedule based on priority, and permit or reject each request. While in the assign mode, IM collects information about motion state of vehicles approaching intersection area and allocates space-time slots for their passage. Distributed approach remove centralized manager and IM can act as coordinator to process and distribute messages. CAVs adjust their speeds or trajectories through self-decision algorithm after receiving messages or form consensus-based crossing schedules after communication. The process of sending, receiving, verifying each other’s status and reaching consensus will lead to more intensive communication burden than centralized way. Platoon with leading CAV is proven to significantly reduce communication load[3], as well as improve the throughout in AIM.

Otto et al analyzed the pros and cons of centralized and distributed method[4] and found they trade-off between computational complexity and performance. Generally speaking, centralized way performs better as global observation and consideration. However, with regard to robustness, it may suffer from high dependencies upon clock synchronicity.
when collecting messages from CAVs and make decisions globally \cite{5} compared with distributed way. Correspondingly, the defects of distributed method consist of probabilities of falling into local optimum restricted to the locality of receptive field. To take advantage of their respective superiority, some studies adopted a hierarchical way for AIM \cite{6} and inspire us hierarchical method has greater potential combined with distributed DRL.

As is well known the technologies of autonomous driving develop with neural networks in computer vision and DRL for planning and decision. While the application of learning-based method in AIM is far less than optimization-based or rule based. Though the internal algorithm has made each CAV an intelligent agent, their experiencing scenes and training standards are distinctive, making the interaction among them unpredictable and unstable. They desire an experienced "navigator" when crossing the intersection involved several conflicting zone. Following the way \cite{7} deployed to enhance safety and efficiency when traffic condition changes in highway, speed advisories are availed by "navigator" to lead CAVs to crossing intersection. These "navigators" collect the collective experiences generated by CAVs in the intersection scene to form "intelligent personality" by training DRL models.

Single agent to navigate all CAVs lacks of learning ability and representation of interactive behavior, while multi-agent DRL encounters curse of dimensionality in AIM. Actors cluster CAVs into multiple personalities to help them coordinate with each other and share memories within the same actor to enrich their experiences and broaden their vision. Moderate actors with lower priority tend to give way to radical actors with higher priority, which helps to prevent deadlocks and game dilemma at intersection. Main contributions of this paper include:

1. Soft Actor-Critic (SAC), an advanced policy-based DRL model, is adopted to behave as "navigator" to guide CAV through intersection by advising speed given in its perspective.
2. A novel multi-agent DRL framework for AIM, ActorRL, is proposed to tackle with interaction of CAVs, by collecting experiences from CAVs to form collective personality of actor, and allocate actors to different CAVs. Actor allocation in ActorRL provides ideas for solving interaction in multi-agent DRL problem.
3. Actor switching mechanism supports CAV to act differently under different situations, reduces unavoidable collisions and performs better in suddenly changing traffic condition.

The rest of the paper is organized as follows. Chapter II reviews relevant literatures in AIM domain, Chapter III illustrates details of ActorRL. Chapter IV describes the experiments and compare ActorRL with baselines. Conclusions and discussions are presented in Chapter V.

2 Literature Review

2.1 Traffic Light and AIM

As a key infrastructure in road networks, traffic light has played the role of intersection conductors for decades and a number of research has been carried out around it. Despite long-term development, traffic lights in many cities around the world still deploy fix-time strategy, where Webster \cite{8} signal setting is one of the most famous method. Besides, various adaptive timing strategies like vehicle-actuate \cite{9}, pressure-based \cite{10, 11} has been launched. As a pressure-based approach, largest-queue-first was proven high robustness by always letting direction with the longest queue pass \cite{12}. The AIM, proposed by Drenseret al. \cite{13}, behaves as self-organized system for CAVs to reduce unnecessary stops and decelerations in predefined traffic signal control. In AIM, intersection area might cover observation zone, optimization zone and control zone, where CAV observes the status and order of others, optimizes their schedules and calculates desired speed, and perform specified actions, respectively \cite{14, 15}. Other division covers queuing zone and acceleration zone \cite{16}. When entering observation zone and optimization zone, request and reply message set is defined differently between centralized and distributed way. In centralized method, four kinds of messages CAV will send, "request" asks for right of way, "adjust" regulates motion status and time schedule, "acknowledge" verifies knowledge about command, and "done" is set when leaving intersection \cite{17}. Correspondingly, IM will send "agree" or "reject" for requests. Decentralized communication applies another set of communication vocabulary, "request" asks for approaching conflicting zone, "approve" agrees with "request" and transit the CAV to crossing state, "interrupt" requests conflicting vehicles to stop and "yield" responses to "interrupt" \cite{18}. Moreover, ad-hoc technique is utilized to overcome the limited receptive field in V2V and shown significant improvements in driving safety at intersections \cite{19}. There are also some signals taking vulnerable traffic participants into account, V-alert combines long-range communication and short-range communication to give driver and pedestrian more time to react and avoid collisions in the intersection area \cite{20}.

In the scenario of 100% CAVs at intersection, AIM is able to make full use of space-time resources to shorten travel delay by allowing the non-conflicting vehicles to enter the intersection simultaneously. While seeing as long-term coexistence of HVs and CAVs, traffic light might still exist for decades. Reports indecate that the market occupation rate of CAVs
will not exceed 90% by 2045 [21]. Thus, the hybrid intersection which focuses on the crossing policy with CAVs and HVs has attracted intensive studies. An initial attempt in hybrid system, FCFS-Light associate First-Come-First-Serve (FCFS) Algorithm with traffic light. Both HVs and CAVs are allowed to cross intersection during green phase, while HVs must stop in red and CAVs are permitted when they neither conflict with existing reservations nor occupy the green or yellow lanes [22]. H-AIM fills the gap of poor performance of the poor performance of FCFS-Light under low CAV penetration by adjusting the rule of no-reservation green lane to no-reservation active green lane [23]. Besides, fuzzy-logic control and reinforcement learning are also leveraged to solve traffic schedules in hybrid intersection [24,25].

### 2.2 Scheduling Policy

The scheduling strategy essentially reorder the entering or leaving sequence of vehicles regarding one or more of the scheduling objectives from safety, efficiency, fairness, environment, and comfort. Algorithms for scheduling are classified into First-In-First-Out (FIFO), heuristic and optimization-based [26]. First-In-First-Out, as the name implies, the earlier approaching vehicles leave the intersection earlier. Optimization-based method formulates AIM as conditional optimization problem and seeks for the global optimum, while heuristic way swap the order between pairs of cars per time. FIFO shows more fairness, optimization-based gain least delay, whereas heuristic approach achieve a trade-off between them [2].

Specifically, many operational planning means are harnessed for AIM. Two methods in linear optimization, Big M method [27] and Brand-and-Bound [28] are introduced to find the solution. Seyed et al. brought in mixed-integer linear programming (MILP) to solve scheduling optimization [29], and then it was also employed in the trajectory selection [30] and analogous jobs includes mixed-integer non-linear programming (MINLP) for collision-free planning [31]. Other approaches involves Lagrange function, Euler-Lagrange equation and inexact Newton method [32,33]. In addition, swarm optimization naturally suits for searching for target solution of multi-agent optimization as it intuitively reflects the problem and save computing time through high parallelism. Along this line, Mehmet et al. present Particle Swarm Optimization [36] and Saust et al. applied max-min ant system [37] in AIM. Apart from aforementioned planning methods, numerous works focus on four branches: tree and search, bidding mechanism and co-utility maximization, game and consensus, and fuzzy logic approach.

Li et al put forward a tree algorithm where each leaf corresponds to an efficient schedule thus each vehicle can seek optimal entry time by traversing the tree [38]. Adaptive belief tree in [39] attempted to give the solution of partially observable Markov decision process, which AIM problem was formulated as considering uncertainty from intentions of human driver and perceived noise from sensors. Choi et al. created a red-black tree representing conflicts and traversed tree for earliest passable time slot [16]. Monte Carlo tree search accelerate the search for target schedule in [40]. Stochastic look-ahead policy based on Monte Carlo tree search was adopted in [41] to find the near optimal solution for CAVs. Other than tree search, window search algorithm assist to adjust velocities when planning trajectories through the intersections [42].

Bidding system has been used for path selection in urban networks [43], then it is extended to resolve conflicts between CAVs at intersection. CAVs bid constantly in auction until defeat its competitors and acquire right of way [44]. Reservations mechanism for vulnerable parities remains proper space-time slot for low-budget CAV to achieve higher flexibility for vehicles joining or leaving platoon [53].
Finally, the fuzzy logic reflects human driver’s perception and decision-making patterns well, thus it has been utilized for the interactions between CAVs and HVs at hybrid intersection or to enhance the robustness amongst CAVs in a few studies [24, 54, 55].

3 Methodology

3.1 Problem Statement

The purpose of managing and scheduling in AIM, is to primarily guarantee the safety of traffic participants and enable vehicles to cross the interweaving area with shortest delay via V2V or V2I communication. Compared to widely studied rule-based scheduling optimization, RL is relatively less deployed in AIM due to challenges lying in the centralized and distributed way. They confront a trade-off between huge joint states and insufficient consideration of interactions which might lead to instability and divergence. Allow for the sparsely coupling between agents, DCL-AIM availed joint Q-values at coordinated states and independent Q-values at independent states in combination [56]. Other works make use of game theory to depict the coordination between agents [57]. While the pair-wise cooperation still faces challenges deploying in large-scale interactive scenarios. Inspired by the coordination among human drivers and crowd, intention inference and action decision are relevant with the personality of drivers and the role they play when confronting, mild driver or driver in non-prioritized states is intended to yield to radical one in the prioritized states. Various roles will differentiate encountering drivers and help to escape from the indeterminate decisions made from the dilemma of symmetric states in single-role RL.

3.2 Multi-agent Reinforcement Learning

In the RL framework, agent interacts with environment by sampling action from strategy \( \pi(a_t|s_t) \) at step \( t \), and acquires corresponding reward \( r_t \) from environment and transfer to next state \( s_{t+1} \). The process keeps shaping agent’s principle of actions until it finds an optimal policy, and it can be formulated as a Partially Observable Markov Decision Process (POMDP) in AIM and is defined by five-variable tuple \( (S, A, P, R, \gamma) \), where state \( S \) and transmission probability \( P \) are given by environment, \( A \) denotes action space of agent, and \( R \) gives feedback of each action, the \( \gamma \) discount future reward to prevent infinite total reward. To deal with huge joint state in centralized deployment, convolutional networks are collaborated to extract spatial features but it cannot fully reflect the interactions between agents. Thus multi-agent RLs (MARL) in distributed way is fit for scheduling in AIM and formulated as multi-agent MDP:

\[
V := (n, S(1), S(2), \ldots, S(n), A(1), A(2), \ldots, A(n), P, R(1), R(2), \ldots, R(n), \gamma)
\]

where \( n \) denotes the number of agents, and \( S_{\text{joint}} = S(1) \times S(2) \times \ldots \times S(n) \), \( A_{\text{joint}} = A(1) \times A(2) \times \ldots \times A(n) \) depict the joint-state and joint-action space respectively. To avoid the curse of dimensionality in state-action space, MARL has developed several frameworks, like independent RL (IRL), fully observable critic, value function factorization, consensus-based cooperation and learn-to-communicate mechanism. Observing and Communicating globally and behaving coordinately prevent each agent from falling into self-centered decision making and reducing the overall utilities. Our method offers hints for applying IRL in the MARL problem with rule-based actor assignment and switching, unnecessary for design of complicated communications. The tuple \( V \) can be condensed into:

\[
V \leftarrow (m, S_1, S_2, \ldots, S_m, A_1, A_2, \ldots, A_m, P, R_1, R_2, \ldots, R_m, \gamma)
\]

where \( m \) represents the number of actors to be allocated, involving a set of agents: \( S_i = S_i^{(1)} = S_i^{(2)} = \ldots = S_i^{(k)}, A_i = A_i^{(1)} = A_i^{(2)} = \ldots = A_i^{(k)}, i = 1, 2, \ldots, m \). The state-action pair \( (S_i^{(j)}, A_i^{(j)}) \) denotes the \( j \)-th agent playing \( i \)-th actor. The five-variable tuple \( (S, A, P, R, \gamma) \) of each actor is a cluster of CAVs belonging to the actor and trained with sharing memories.

Figure 1: Interactions between CAVs as different actors
### 3.3 Soft Actor Critic

As an advanced off-policy actor-critic algorithm, Soft Actor Critic is capable of exploring more actions with entropy maximization term rather than gets stuck in a local optimal trajectory. And the policy updated in each step is proven better than before [58]. The property enables the agent improve their ability stably, suitable for training actor agents.

The observations of each CAV are given under its own relative coordinates, involving driving status of surrounding vehicles with receptive field with a radius of 20m. The receptive field is put into a $40 \times 40$ square centered on the ego car and further divided into $20 \times 20$ grids with side length of a grid 2m as shown in Fig 2. Given $H^{(j)}_i$ the $j$-th CAV as $i$-th actor, its state at step $t$ is expressed as:

$$s^{(j)}_i(t) = (x^{(j)}_{i,0}, y^{(j)}_{i,0}, v^{(j)}_{i,0}, \text{occup}^{(j), \text{rel}}_{i,t}, \text{speed}^{(j), \text{rel}}_{i,x}, \text{speed}^{(j), \text{rel}}_{i,y})$$ (3)

Where $s^{(j)}_i(t)$ is a 1203-dim vector, $x^{(j)}_{i,0}, y^{(j)}_{i,0}, v^{(j)}_{i,0}$ characterize the current motion status of ego CAV in absolute coordinates, meaning horizontal and vertical position and speed. Each of $\text{occup}^{(j), \text{rel}}_{i,t}, \text{speed}^{(j), \text{rel}}_{i,x}, \text{speed}^{(j), \text{rel}}_{i,y}$ is 400-dim vector reshaped from the $20 \times 20$ receptive matrix. Elements in $\text{occup}^{(j), \text{rel}}_{i,t}$ are made up of 0-1 binary values representing whether each grid is occupied. When the center of a grid falls inside the outline of a vehicle, then the corresponding element is 1 otherwise 0. $\text{speed}^{(j), \text{rel}}_{i,x}$ and $\text{speed}^{(j), \text{rel}}_{i,y}$ denote the horizontal and vertical speed of adjacent vehicle in the relative coordinates of ego vehicle.

Restricted in the action space $[a_{\text{min}}, a_{\text{max}}]$, the action of $H^{(j)}_i$ at each step $t$ is randomly generated or is generated from neural networks.

$$a^{(j)}_{i,t} = \begin{cases} \text{random}(a_{\text{min}}, a_{\text{max}}), & \text{if } t \leq \text{start time} \\ f_\phi_i(\epsilon^{(j)}_{i,t}; s^{(j)}_{i,t}) & \text{if } t > \text{start time} \end{cases}$$ (4)

Where $\phi_i$ is hyper-parameter of actor networks for $i$-th actor and $\epsilon^{(j)}_{i,t}$ denotes Gaussian noise, $s^{(j)}_{i,t}$ is the current state. The action of each CAV is defined as the expected speed within space action $[v_{\text{min}}, v_{\text{max}}]$ at next moment. Taking smoothness of action into account, the speed of vehicle at next simulated timestep is restricted to a variation range of $1m/s^2$ under normal circumstances and $2m/s^2$ when emergency.

$$\max(v_t - 1, 0) \leq v^{\text{normal}}_{i,t+1} \leq v_t + 1$$ (5)

$$\max(v_t - 2, 0) \leq v^{\text{emergency}}_{i,t+1} \leq v_t + 2$$ (6)
The goal of ActorRL is to seek for optimal collision-free passing schedule for each vehicle, thus the value of state-action pair \((s_{i,t}, a_{i,t})\) is defined as:

\[
\text{val} \left( s_{i,t}^{(j)}, a_{i,t}^{(j)} \right) = \begin{cases} 
-1, & \text{if } H_{i}^{(j)} \text{ is still in the intersection} \\
-3, & \text{if } H_{i}^{(j)} \text{ gets into dangerous situation} \\
-5, & \text{if } H_{i}^{(j)} \text{ involves in collisions} \\
0, & \text{if } H_{i}^{(j)} \text{ leaves the intersection with collisions before} \\
10, & \text{if } H_{i}^{(j)} \text{ leaves the intersection without collision}
\end{cases}
\]

(7)

The reward of \((s_{i,t}^{(j)}, a_{i,t}^{(j)})\) is defined as sum-up of values on the following tracks of agent until it leaves the intersections.

\[
r \left( s_{i,t_0}^{(j)}, a_{i,t_0}^{(j)} \right) = \sum_{t=t_0}^{T_{\text{end}}} \text{val} \left( s_{i,t}^{(j)}, a_{i,t}^{(j)} \right)
\]

(8)

Where \(T_{\text{end}}\) denotes the steps agent leaving intersection from this moment.

Each actor is dynamically made up with cluster of \(k\) vehicles at intersection area \(H_i^{(1)}, H_i^{(2)}, \ldots, H_i^{(k)}\) at given moment, consisting a SAC agent and a memory palace \(D_i\), where \(D_i\) updates in the following way:

\[
D_i = D_i \cup \left\{ \left( s_{i,t}^{(j)}, a_{i,t}^{(j)}, r \left( s_{i,t}^{(j)}, a_{i,t}^{(j)} \right), s_{i,t+1}^{(j)} \right) \right\}
\]

(9)

Where \(\delta\) is a signal indicating whether agent has finished its task for a vehicle.

Each SAC agent involves a Value Network, a Q Network and a Policy Network. The Value network gives the value estimations of states with objective function:

\[
J_V \left( \psi_i \right) = \mathbb{E}_{s_t \sim D_i} \left[ \frac{1}{2} \left( V_{\psi_i}(s_t) - \mathbb{E}_{a_t \sim \pi_{\phi_i}} [Q_{\theta_i}(s_t, a_t) - \log \pi_{\phi_i}(a_t | s_t)] \right)^2 \right]
\]

(10)

Where \(V_{\psi_i}(s_t), Q_{\theta_i}(s_t, a_t), \pi_{\phi_i}(a_t | s_t)\) are the values yielded from Value Network, Q Network and Policy Network in \(i\)-th actor, and \(\psi_i, \theta_i, \phi_i\) denote the hyper-parameters of those networks, respectively. \((s_t, a_t)\) is a state-action pair sampled from memory \(D_i\). And the objective function of Q Network for \(i\)-th actor is given as:

\[
J_Q \left( \theta_i \right) = \mathbb{E}_{(s_t, a_t) \sim D_i} \left[ \frac{1}{2} \left( Q_{\theta_i}(s_t, a_t) - (r(s_t, a_t) + \gamma \mathbb{E}_{s_{t+1} \sim p} [V_{\psi_i}(s_{t+1})]) \right)^2 \right]
\]

(11)

Where \(\psi_i\) is an exponentially moving average of value network weights updated with \(\psi_i \leftarrow (1 - \tau)\psi_i + \tau \hat{\psi}_i\) to stabilize the training. Further, the objective function of policy network for \(i\)-th actor is shown as:

\[
J_\pi \left( \phi_i \right) = \mathbb{E}_{s_t \sim D_i, \epsilon_t \sim \mathcal{N}} \left[ \log \pi_{\phi_i}(f_{\phi_i}(\epsilon_t; s_t) \mid s_t) - Q_{\theta_i}(s_t, f_{\phi_i}(\epsilon_t; s_t)) \right]
\]

(12)

Following the work in [58], automated tuning of temperature parameter are utilized during training.

### 3.4 Actor allocation

For the convenience of expression, The notations that will be used later are given in the table.

The connection is a predefined trajectory from entry-lane to exit-lane, and all conflicting points between different connections makes up \(N_v\)-dim conflicting list.

We divide the area of IM into three zones: allocating zone (AZ), preparing zone (PZ) and controlling zone (CZ). When reaching the AZ, CAV will keep trying to be allocated as an actor before leaving this zone, the allocation process is built upon reservation table. If \(i\)-th actor is allocated, CAVs will slow down or speed up to \(v_{i,mid}\) before entering the CZ. CAV takes actions from SAC model to cross safely in CZ. The process is shown in Figure.

In the framework of ActorRL, We enumerate 9 kinds of actors with different turning intentions and driving aggressiveness as shown in table. The top three actors under category of straight and left-turn are allocated in AZ and the last actor takes control when allocated actor cannot behave properly within specified action span. The action spans of top three actors are selected to make sure each actor can be allocated enough CAVs to acquire experiences.

Considering the occupation ratio in the spatial-temporal crossing trajectory of vehicle, the principle of actor allocation is illustrated: drive fast when safety otherwise slowly, and if all actors cannot cross safely, CAV will undertake
Table 1: Notations

| Notation | Representation |
|----------|----------------|
| $N_c$    | The number of conflicting points at intersection |
| $t$      | Current time step |
| $T$      | The number of time steps in the reservation table |
| $\Delta t_i$ | The temporal resolution of reservation table for $i$-th actor |
| $h^{(j)}$ | The $j$-th vehicle in the vehicle list |
| $v^{(j)}(t)$ | Current velocity of $h^{(j)}$ |
| $v_{i,mid}$ | The median velocity in the action space of $i$-th actor |
| $v_{i,high}$ | The highest velocity in the action space $i$-th actor |
| $v_{i,low}$ | The lowest velocity in the action space of $i$-th actor |
| $Z^{(j)}(t)$ | The zone which $h^{(j)}$ is in currently |
| $q_m$ | The $m$-th conflicting point in conflicting points |
| $Q^{(j)}$ | A list of conflicting points passed by $h^{(j)}$ |
| $N_r$ | The number of actors can be allocated to CAV under its turning intention |
| $O^{(j)}$ | Average occupation ratio list of $h^{(j)}$ as $i$-th actor |
| $t^{(j)}$ | The arrival time of $h^{(j)}$ to stop line with current speed |
| $t^{(j)}_i$ | The average arrival time of $h^{(j)}$ as $i$-th actor to stop line |
| $t^{(j)}_{i,e}(q_m)$ | The earliest arrival time of $h^{(j)}$ as $i$-th actor to $q_m$ from this moment. |
| $t^{(j)}_{i,l}(q_m)$ | The latest arrival time of $h^{(j)}$ as $i$-th actor to $q_m$ from this moment. |
| $t^{(j)}_{e}(q_m)$ | The earliest arrival time of HV or unallocated CAV $h^{(j)}$ to $q_m$ from this moment. |
| $t^{(j)}_{l}(q_m)$ | The latest arrival time of HV or unallocated CAV $h^{(j)}$ to $q_m$ from this moment. |
| $M_{base}$ | The static reservation table with size of $N_r \times N_c \times T$ |
| $M_{dyn}$ | The dynamic reservation table with size of $N_r \times N_c \times T$ |
| $Y^{(j)}$ | The actor allocated to $h^{(j)}$ in $\beta$ round |
| $L^{(j)}$ | The lane $h^{(j)}$ drives on |
| $L^{(j)}_{cand}$ | The candidate lane $h^{(j)}$ able to drive on |

Table 2: Classification of actors

| Actor ID | Turning intention | Driving aggressiveness | Velocity span |
|----------|-------------------|------------------------|---------------|
| 1        | High              | [10,12]                |               |
| 2        | Straight          | Medium [6,10]          |               |
| 3        | Low               | [4,6]                  |               |
| 4        | Safe              | [0,4]                  |               |
| 5        | High              | [10,12]                |               |
| 6        | Left-turn         | Medium [6,10]          |               |
| 7        | Low               | [4,6]                  |               |
| 8        | Safe              | [0,4]                  |               |
| 9        | Right-turn        | Normal [0,10]          |               |

Lane-changing or slow down even stop to wait a chance. To this end, we build reservation table with size of $N_r \times N_c \times T$ and factorize it into static reservation table $M_{base}$ and dynamic reservation table $M_{dyn}$. $M_{base}$ gives static reservation involving all vehicles except CAV to be allocated in this allocation stage. The dynamic table $M_{dyn}$ depicts the reservations requested by allocating CAV and is updated dynamically during the process of actor allocation. The earliest and latest arriving time to conflicting point $q_m$ is given as:

\[
\begin{align*}
    t_{i,e}^{(j)}(q_m) &= \frac{2d_p}{v_{0} + v_{i,mid}} + \frac{(d_m - d_p)}{v_{i,low}} \\
    t_{i,l}^{(j)}(q_m) &= \frac{2d_p}{v_{0} + v_{i,mid}} + \frac{(d_m - d_p)}{v_{i,high}}
\end{align*}
\] (13)
Where $v_0$ is the current speed of vehicle, $d_p$ denotes the preparing distance for agent to speed up or slow down to $v_{i,mid}$ and $d_m$ denotes the distance from conflicting point $q_m$ at this step. Similarly, the volatility of the speed of HVs is calculated by:

$$\begin{align}
\tau_e(q_m) &= d_m v_0 + \Delta v \\
\tau_l(q_m) &= d_m \max(v_0 - \Delta v, 1)
\end{align}$$

then the $M_{base}$ or $M_{dyn}$ will updated with perspective from each $i-$th actor:

$$M_*(i, m, t_{start} : t_{end}) = 1$$

Where * denotes 'base' or 'dyn' and $t_{start}$ and $t_{end}$ can be expressed as for $M_{base}$:

$$\begin{align}
t_{start} &= \text{clip}(\text{int} \left( \frac{\tau_e(q_m)}{\Delta t_i} \right), (0, T)) \\
t_{end} &= \text{clip}(\text{int} \left( \frac{\tau_l(q_m)}{\Delta t_i} \right), (0, T))
\end{align}$$

or for the $M_{dyn}$:

$$\begin{align}
t_{start} &= \text{clip}(\text{int} \left( \frac{\tau_e(q_m)}{\Delta t_i} \right), (0, T)) \\
t_{end} &= \text{clip}(\text{int} \left( \frac{\tau_l(q_m)}{\Delta t_i} \right), (0, T))
\end{align}$$

Then occupation ratio as $i-$th actor is expressed as:

$$O^{(j)}_i = \frac{\sum_{q_m \in Q^{(j)}} M(i, q_m, t_{end} : t_{start})}{\sum_{q_m \in Q^{(j)}} (t_{end} - t_{start})}$$

Then take case of driving straight as an example, the flow chart for actor selection is shown as [4]

We define $H := [h^{(1)}, h^{(2)}, \ldots, h^{(k)}]$ as list of allocating CAV at this stage, and some of them finish the allocation process, written as $H' := [h'^{(1)}, h'^{(2)}, \ldots, h'^{(u)}]$, a dynamic dictionary $D_t$ records the occupied spatial-temporal slots of allocating CAVs during allocating process. The pseudo-code of updating $M_{dyn}$ is given in algorithm [1]
Algorithm 1: Updating Dynamic reservation table $M_{dyn}$

**Inputs**: $H$, $M_{dyn}$, $D_t$

**Outputs**: $M_{dyn}$, $D_t$

1. for $j = 1, 2, ..., k$ do
   2. $i = $ actor allocated to $h^{(j)}$ according to Fig4
      3. for $q_m$ in $Q^{(j)}$ do
         4. if $(h^{(j)}, q_m)$ in $D_t$ then
            5. $t_{e,in} = D_t((h^{(j)}, q_m))$
               6. for $i' = 1, 2, ..., N_r$ do
                  7. $t_{start}, t_{end} \leftarrow \text{Equation (17)}$
                     8. $M_{dyn}(i', q_m, t_{start} : t_{end}) - 1$
            9. end
      10. end
      11. $t_{e,in}, t_{i', in}^{(j)} \leftarrow \text{Equation (14)}$
      12. $D_t((h^{(j)}, q_m)) \leftarrow (t_{i', in}^{(j)}, t_{i', in}^{(j)})$
      13. for $i' = 1, 2, ..., N_r$ do
           14. $t_{start}, t_{end} \leftarrow \text{Equation (17)}$
              15. $M_{dyn}(i', q_m, t_{start} : t_{end}) + 1$
         16. end
      17. end
  18. end
  19. return $M_{dyn}$, $D_t$

There is a 4-round allocation before the actor of a CAV is determined. The purpose of multi-round assignment is to guarantee the stability of allocation results and the allocation sequence of allocating CAV follows the principle of FCFS. In first round, the arrival time of $h^{(j)}$ is yielded from its current speed and get an actor list $Y_1 := [Y_1^{(1)}, Y_1^{(2)}, \ldots, Y_1^{(k)}]$ which denotes the initially allocated actor in this round. Then the allocation sequence is re-calculated based on $Y_1$ and re-do the actor allocation based on $[\text{4}]$ if the results of first 2 round are identical then the allocation done, otherwise the allocation is re-calculated based on $Y_2$ and re-do the process, but the difference in this round is the CAVs finishing allocation will no more participate in this round. Finally, in the last round, lane-changing, slow-down or speed-up will be undertaken by unassigned CAVs to seek chances to become actors. The flow chart is shown as Fig 6

After the actor allocation completed, the CAV will accelerate or decelerate to $v_{i,mid}$ gradually within the preparing zone. The spatial relations between ego CAV and front and rear vehicles are considered in this process. If there is no proper distance to accelerate to the target speed, CAV will slow down to expand the heading gap or overtake front cars under the premise of safety.

When crossing the intersection, the actor who leads the CAV will switched under 3 conditions:

1. Action from action space of current actor inevitably leads to collisions or dangerous situation.
2. The traffic conditions at intersection change significantly and getting stuck into unexpected congestion.
3. The crossing schedules of vehicles in close proximity in space and time change, causing the determined by allocation rules different from the actor before
If existing any of condition above, CAV $h^{(j)}$ will re-do actor allocation if $Z^{(j)}(t) = AZ$ or $PZ$, otherwise it will lower its action level until it finds a safe optimal actions. For example, if CAV with action span of $[10, 12]$ confronts foreseeable conflicts, it will seek action from $[6, 10]$ then $[4, 6]$ and finally $[0, 4]$. The reward at the action-switching step is given as:

$$r_{t}^{role1} = V^{role2}(s_t)$$  \hspace{1cm} (19)

Where $role1$ denotes the actor before switching and $role2$ after switching. $V^{role2}$ represents the value evaluated by the Value Network of $role2$.

### 3.5 Human Driver

The behavioral model of human drivers is adopted intelligent drive model (IDM), which can be expressed as:

$$a_{IDM} = a_{\text{max}} \left[ 1 - \left( \frac{v^{(k)}(t)}{v_0^{(k)}} \right)^3 - \frac{s^{(k,k-1)}(t)}{s^{(k,k-1)}(t)} \left( v^{(k)}(t), \Delta v^{(k,k-1)}(t) \right) \right]$$  \hspace{1cm} (20)

$$s^{(k,k-1)}(v^{(k)}, \Delta v^{(k,k-1)}) = s_0 + v^{(k-1)}T + \frac{v^{(k-1)}\Delta v^{(k-1)}}{2\sqrt{a_{\text{max}}b}}$$  \hspace{1cm} (21)

Where $a_{\text{max}}$ denotes the maximum acceleration of ego vehicle, $v^{(k)}$ denotes the current velocity and $v_0^{(k)}$ represents the expected velocity of ego car. $s^{(k,k-1)}$ and $s^{(k,k-1)}_e$ means the current heading and expected heading between ego car and front car. $b$ is for comfortable deceleration.

### 4 Experiments

#### 4.1 Description of Experiments

The simulations are conducted in widespread microscopic traffic simulator, SUMO. The test scenarios are configured as 4-way-8-lane intersection without traffic signal. The trajectory of vehicle’s movements follows the road markings. The simulation step is set to 0.1s. Some detailed parameters of simulation are given as table 3.

---

Figure 5: Four round actor allocation for CAV
Table 3: Details of Parameters

| Parameters | Values   | Explanations                      |
|-----------|----------|-----------------------------------|
| $l_v$     | 5.0m     | The length of vehicles            |
| $w_v$     | 1.8m     | The width of vehicles             |
| $a_{max}$ | 4m/s$^2$ | The maximum acceleration of vehicles |
| $a_{max}$ | 3m/s$^2$ | The maximum deceleration of vehicles |
| $d_{AZ}$  | [30, 50]m| The distance range from allocation zone to stop line |
| $d_{PZ}$  | [10, 30]m| The distance range from prepare zone to stop line |
| $v_{max}$ | 20m/s    | The maximum speed of vehicles in the scenarios |
| $N_{epoch}$ | 100     | The maximum number of training epochs |
| $M_{mem}$ | 100000   | The size of replaying memories of each SAC |
| $t_{action}$ | 1.0s    | The interval of agent’s action step |
| $B$       | 256      | The batch size of data input into neural networks |
| $\gamma$  | 0.99     | The discount factor for reward    |
| $\alpha$  | 0.2      | Temperature parameter for importance of entropy |
| $lr$      | 0.0003   | Learning rate of the neural networks |

4.2 Performance Evaluation

We select the metrics of experiments from aspects of safety, efficiency, and fuel consumption:

1. $N_{col} :=$ the number of collisions per hour.
2. $t_{cross} :=$ the average crossing time from vehicles entering CZ to leaving the intersection. The boundary of CZ is defined 10m away from the stop line.
3. $std_t :=$ the standard deviation of $t_{cross}$. The metric reflects the fairness of right of road for different vehicles.
4. $F :=$ the average fuel consumption per vehicle.

4.3 Method Comparison

Five methods are employed for comparison with the proposed model.

1. Fixed signal timing control: pre-defined signal phases with fixed order and duration change periodically in a round-robin manner. The cycle of signal is set to 60s in this experiment.
2. Longest-Queue-First (LQF) timing signal: As a robust method introduced in literature review, LQF permits the vehicles in the directions with longest queue to cross intersection first.
3. FCFS with virtual traffic light (VTL): The CAVs follows the principle of FCFS, and HVs follow its front CAV to cross the intersection.
4. FCFS with virtual traffic light (platoon): On the basis of FCFS with VTL, platoons are formed based on the proximity of vehicles’ crossing time to enhance the efficiency, the maximum size of platoon is constant and defined as 8 in our experiments.
5. Distributed SAC without actor allocation: As a comparison of the proposed method, single SAC model is employed to guide each CAV to cross the intersection.

4.4 Results and Analysis

|                     | $C$ | $t_{cross}$(s) | $std_t$(s) | $F$(ml/veh) |
|---------------------|-----|----------------|------------|-------------|
| Fixed time signal   | 0   | 32.96          | 42.22      | 48.12       |
| LQF signal          | 0   | 14.8           | 11.9       | 28.95       |
| FCFS-VTL            | 0   | 31.06          | 23.91      | 35.65       |
| FCFS-VTL(Platoon)   | 0   | 29.56          | 28.54      | 37.74       |
| Distributed RL      | 33  | 11.42          | 2.61       | 11.29       |
| Proposed method     | 0   | 9.63           | 2.33       | 12.95       |
The results show that actor allocation mechanism can significantly improve the safety for reinforcement learning in hybrid intersection management, and also shorten the crossing time. Higher fuel consumption might be caused by more frequent operation caused by complicated scenarios. But to sum up, Actor-RL performs better than the baseline model here.

5 Conclusion

This paper develops a novel framework for interactions of multi agents in hybrid intersection. By playing different actors and switching actors under varying situation, CAVs is able to find schedule to cross intersection safely by itself. The experiment is conducted in a single intersection and the proposed model performs well in this scenarios. Further work can extend this cooperation paradigm to more scenarios like multi-intersection coordination. And more method for allocating actors can be explored in the future.

References

[1] Chaoyi Chen, Jiawei Wang, Qing Xu, Jianqiang Wang, and Keqiang Li. Mixed platoon control of automated and human-driven vehicles at a signalized intersection: dynamical analysis and optimal control. *Transportation Research Part C: Emerging Technologies*, 127:103138, 2021.

[2] Mohammad Khayatian, Mohammadreza Mehrabian, Edward Andert, Rachel Dedinsky, Sarthake Choudhary, Yingyan Lou, and Aviral Shirvastava. A survey on intersection management of connected autonomous vehicles. *ACM Transactions on Cyber-Physical Systems*, 4(4):1–27, 2020.

[3] Qiu Jin, Guoyuan Wu, Kanok Boriboonsomsin, and Matthew Barth. Platoon-based multi-agent intersection management for connected vehicle. In *16th international ieee conference on intelligent transportation systems (itsc 2013)*, pages 1462–1467. IEEE, 2013.

[4] John S Otto and Fabián E Bustamante. Distributed or centralized traffic advisory systems-the application’s take. In *2009 6th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, pages 1–10. IEEE, 2009.

[5] Edward Andert, Mohammad Khayatian, and Aviral Shirvastava. Crossroads: Time-sensitive autonomous intersection management technique. In *Proceedings of the 54th Annual Design Automation Conference 2017*, pages 1–6, 2017.
[6] Jean Gregoire and Emilio Frazzoli. Hybrid centralized/distributed autonomous intersection control: Using a job scheduler as a planner and inheriting its efficiency guarantees. In 2016 IEEE 55th Conference on Decision and Control (CDC), pages 2549–2554. IEEE, 2016.

[7] Tarek Ghoul and Tarek Sayed. Real-time safety optimization of connected vehicle trajectories using reinforcement learning. Sensors, 21(11):3864, 2021.

[8] Fo Vo Webster. Traffic signal settings. Technical report, 1958.

[9] FG Tyack. Street traffic signals, with particular reference to vehicle actuation. Journal of the Institution of Electrical Engineers, 82(494):125–154, 1938.

[10] Jean Gregoire, Emilio Frazzoli, Arnaud de La Fortelle, and Tichakorn Wongpiromsarn. Back-pressure traffic signal control with unknown routing rates. IFAC Proceedings Volumes, 47(3):11332–11337, 2014.

[11] Azzedine Boukerche, Dunhao Zhong, and Peng Sun. A novel reinforcement learning-based cooperative traffic signal system through max-pressure control. IEEE Transactions on Vehicular Technology, 2021.

[12] Richard Wunderlich, Cuibi Liu, Itamar Elhanany, and Tom Urbanik. A novel signal-scheduling algorithm with quality-of-service provisioning for an isolated intersection. IEEE Transactions on Intelligent Transportation Systems, 9(3):536–547, 2008.

[13] Kurt Dresner and Peter Stone. Multiagent traffic management: A reservation-based intersection control mechanism. In Autonomous Agents and Multiagent Systems, International Joint Conference on, volume 3, pages 530–537. IEEE Computer Society, 2004.

[14] Peiquan Lin, Jiahui Liu, Peter J Jin, and Bin Ran. Autonomous vehicle-intersection coordination method in a connected vehicle environment. IEEE Intelligent Transportation Systems Magazine, 9(4):37–47, 2017.

[15] Yougang Bian, Shengbo Eben Li, Wei Ren, Jianqiang Wang, Keqiang Li, and Henry X Liu. Cooperation of multiple connected vehicles at unsignalized intersections: Distributed observation, optimization, and control. IEEE Transactions on Industrial Electronics, 67(12):10744–10754, 2019.

[16] Kurt Dresner and Peter Stone. Sharing the road: Autonomous vehicles meet human drivers. In Ijcai, volume 7, pages 1263–1268, 2007.

[17] Shunsuke Aoki and Ragunathan Rajkumar. Dynamic intersections and self-driving vehicles. In 2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPS), pages 320–330. IEEE, 2018.

[18] Stefan Joerer, Michele Segata, Bastian Bloessl, Renato Lo Cigno, Christoph Sommer, and Falko Dressler. To crash or not to crash: Estimating its likelihood and potentials of beacon-based ivc systems. In 2012 IEEE Vehicular Networking Conference (VNC), pages 25–32. IEEE, 2012.

[19] Chunxiao Li and Shigeru Shimamoto. Multi-agent intersection management for connected vehicles. In 2012 International Conference on Connected Vehicles and Expo (ICCVE), pages 185–190. IEEE, 2012.

[20] Guni Sharon and Peter Stone. A protocol for mixed autonomous and human-operated vehicles at intersections. In International Conference on Autonomous Agents and Multiagent Systems, pages 151–167. Springer, 2017.

[21] Enrique Onieva, Unai Hernández-Jayo, Idoia De-la Iglesia, and Jagoba Perez. V-alert: Description and validation of a vulnerable road user alert system in the framework of a smart city. Sensors, 15(8):18480–18505, 2015.

[22] Kurt M Dresner and Peter Stone. Multiagent traffic management: A reservation-based intersection control mechanism. In Autonomous Agents and Multiagent Systems, International Joint Conference on, volume 3, pages 530–537. IEEE Computer Society, 2004.

[23] Guni Sharon and Peter Stone. A protocol for mixed autonomous and human-operated vehicles at intersections. In International Conference on Autonomous Agents and Multiagent Systems, pages 151–167. Springer, 2017.

[24] Enrique Onieva, Unai Hernández-Jayo, Eneko Osaba, Asier Perallos, and Xiao Zhang. A multi-objective evolutionary algorithm for the tuning of fuzzy rule bases for uncoordinated intersections in autonomous driving. Information Sciences, 321:14–30, 2015.

[25] Duy Quang Tran and Sang-Hoon Bae. Proximal policy optimization through a deep reinforcement learning framework for multiple autonomous vehicles at a non-signalized intersection. Applied Sciences, 10(16):5722, 2020.

[26] Ashkan Gholamhosseini and Jochen Seitz. A comprehensive survey on cooperative intersection management for heterogeneous connected vehicles. IEEE Access, 2022.

[27] Qiu Jin, Guoyuan Wu, Kanok Boriboonsomsin, and Matthew Barth. Multi-agent intersection management for connected vehicles using an optimal scheduling approach. In 2012 International Conference on Connected Vehicles and Expo (ICCVE), pages 185–190. IEEE, 2012.
[28] Kaidi Yang, S Ilgin Guler, and Monica Menendez. Isolated intersection control for various levels of vehicle technology: Conventional, connected, and automated vehicles. *Transportation Research Part C: Emerging Technologies*, 72:109–129, 2016.

[29] Seyed Alireza Fayazi and Ardalan Vahidi. Mixed-integer linear programming for optimal scheduling of autonomous vehicle intersection crossing. *IEEE Transactions on Intelligent Vehicles*, 3(3):287–299, 2018.

[30] Muting Ma and Zhixia Li. A time-independent trajectory optimization approach for connected and autonomous vehicles under reservation-based intersection control. *Transportation Research Interdisciplinary Perspectives*, 9:100312, 2021.

[31] Amir Mirheli, Mehrdad Tajalli, Leila Hajibabaie, and Ali Hajbabaie. A consensus-based distributed trajectory control in a signal-free intersection. *Transportation research part C: emerging technologies*, 100:161–176, 2019.

[32] Fethi Belkhouche. Collaboration and optimal conflict resolution at an unsignalized intersection. *IEEE Transactions on Intelligent Transportation Systems*, 3(3):287–299, 2018.

[33] Andreas A Malikopoulos and Liuhui Zhao. A closed-form analytical solution for optimal coordination of connected and automated vehicles. *In 2019 American control conference (ACC)*, pages 3599–3604. IEEE, 2019.

[34] Yunpeng Wang, Pinlong Cai, and Guangquan Lu. Cooperative autonomous traffic organization method for connected automated vehicles in multi-intersection road networks. *Transportation research part C: emerging technologies*, 111:458–476, 2020.

[35] Yuning Jiang, Mario Zanon, Robert Hult, and Boris Houska. Distributed algorithm for optimal vehicle coordination at traffic intersections. *IFAC-PapersOnLine*, 50(1):11577–11582, 2017.

[36] Mehmet Ali Guney and Ioannis A Raptis. Scheduling-based optimization for motion coordination of autonomous vehicles at multilane intersections. *Journal of Robotics*, 2020, 2020.

[37] Falko Saust, Jorn Marten Wille, and Markus Maurer. Energy-optimized driving with an autonomous vehicle in urban environments. In *2012 IEEE 75th Vehicular Technology Conference (VTC Spring)*, pages 1–5. IEEE, 2012.

[38] Li Li and Fei-Yue Wang. Cooperative driving at blind crossings using intervehicle communication. *IEEE Transactions on Vehicular Technology*, 55(6):1712–1724, 2006.

[39] Constantin Hubmann, Marvin Becker, Daniel Althoff, David Lenz, and Christoph Stiller. Decision making for autonomous driving considering interaction and uncertain prediction of surrounding vehicles. *In 2017 IEEE Intelligent Vehicles Symposium (IV)*, pages 1671–1678. IEEE, 2017.

[40] Huile Xu, Yi Zhang, Li Li, and Weixia Li. Cooperative driving at unsignalized intersections using tree search. *IEEE Transactions on Intelligent Transportation Systems*, 21(11):4563–4571, 2019.

[41] Amir Mirheli, Leila Hajibabaie, and Ali Hajbabaie. Development of a signal-head-free intersection control logic in a fully connected and autonomous vehicle environment. *Transportation Research Part C: Emerging Technologies*, 92:412–425, 2018.

[42] Bing Liu, Qing Shi, Zhuoyue Song, and Abdellaker El Kamel. Trajectory planning for autonomous intersection management of connected vehicles. *Simulation Modelling Practice and Theory*, 90:16–30, 2019.

[43] Matteo Vasirani and Sascha Ossowski. A computational market for distributed control of urban road traffic systems. *IEEE Transactions on Intelligent Transportation Systems*, 12(2):313–321, 2011.

[44] Matteo Vasirani and Sascha Ossowski. A market-inspired approach for intersection management in urban road traffic networks. *Journal of Artificial Intelligence Research*, 43:621–659, 2012.

[45] Dustin Carlino, Stephen D Boyles, and Peter Stone. Auction-based autonomous intersection management. In *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pages 529–534. IEEE, 2013.

[46] Guang Chen and Kyoung-Don Kang. Win-fit: Efficient intersection management via dynamic vehicle batching and scheduling. In *2015 International Conference on Connected Vehicles and Expo (ICCVE)*, pages 263–270. IEEE, 2015.

[47] Muhammed O Sayin, Chung-Wei Lin, Shinichi Shiraishi, Jiajun Shen, and Tamer Başar. Information-driven autonomous intersection control via incentive compatible mechanisms. *IEEE Transactions on Intelligent Transportation Systems*, 20(3):912–924, 2018.

[48] Noam Buckman, Alyssa Pierson, Wilko Schwarting, Sertac Karaman, and Daniela Rus. Sharing is caring: Socially-compliant autonomous intersection negotiation. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6136–6143. IEEE, 2019.
[49] Mohammed Elhenawy, Ahmed A Elbery, Abdallah A Hassan, and Hesham A Rakha. An intersection game-theory-based traffic control algorithm in a connected vehicle environment. In 2015 IEEE 18th international conference on intelligent transportation systems, pages 343–347. IEEE, 2015.

[50] Yalda Rahmati and Alireza Talebpour. Towards a collaborative connected, automated driving environment: A game theory based decision framework for unprotected left turn maneuvers. In 2017 IEEE Intelligent Vehicles Symposium (IV), pages 1316–1321. IEEE, 2017.

[51] Haoran Wei, Lena Mashayekhy, and Jake Papineau. Intersection management for connected autonomous vehicles: A game theoretic framework. In 2018 21st International Conference on Intelligent Transportation Systems (ITSC), pages 583–588. IEEE, 2018.

[52] Bigi Varghese Philip, Tansu Alpcan, Jiong Jin, and Marimuthu Palaniswami. Distributed real-time iot for autonomous vehicles. IEEE Transactions on Industrial Informatics, 15(2):1131–1140, 2018.

[53] Jeroen C Zegers, Elham Semsar-Kazerooni, Jeroen Ploeg, Nathan Van De Wouw, and Henk Nijmeijer. Consensus-based bi-directional cacc for vehicular platooning. In 2016 American Control Conference (ACC), pages 2578–2584. IEEE, 2016.

[54] Magdy M Abdelhameed, Mohamed Abdelaziz, S Hammad, and Omar M Shehata. A hybrid fuzzy-genetic controller for a multi-agent intersection control system. In 2014 international conference on engineering and technology (ICET), pages 1–6. IEEE, 2014.

[55] Magdy M Abdelhameed, Mohamed Abdelaziz, S Hammad, and Omar M Shehata. Development and evaluation of a multi-agent autonomous vehicles intersection control system. In 2014 International Conference on Engineering and Technology (ICET), pages 1–6. IEEE, 2014.

[56] Yuanuyuan Wu, Haipeng Chen, and Feng Zhu. Dcl-aim: Decentralized coordination learning of autonomous intersection management for connected and automated vehicles. Transportation Research Part C: Emerging Technologies, 103:246–260, 2019.

[57] Nan Li, Yu Yao, Ilya Kolmanovsky, Ella Atkins, and Anouck R Girard. Game-theoretic modeling of multi-vehicle interactions at uncontrolled intersections. IEEE Transactions on Intelligent Transportation Systems, 2020.

[58] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In International conference on machine learning, pages 1861–1870. PMLR, 2018.