A Scheduling Algorithm for Pass-through of Connected and Automated Vehicle with Different Priorities in Non-signalized Intersection

1st Lujun Wang
School of Mechanical Engineering
Dalian University of Technology
Dalian, China
wanglujun@mail.dlut.edu.cn

2nd Yanjun Shi*
School of Mechanical Engineering
Dalian University of Technology
Dalian, China
syj@ieee.org

3rd Xiangjie Xiao
School of Mechanical Engineering
Dalian University of Technology
Dalian, China
xiangjiexiao@mail.dlut.edu.cn

4th Xuyang Cao
School of Mechanical Engineering
Dalian University of Technology
Dalian, China
saner@126.com

5st Fuzheng Qu
School of Mechanical Engineering
Dalian University of Technology
Dalian, China
fzqu@dlut.edu.cn

2021 IEEE International Conference on Smart Internet of Things (SmartIoT) | 978-1-6654-4511-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/SmartIoT52359.2021.00034

***Abstract*— Non-signalized intersections have received much attention in various research with the development of Vehicle-to-Everything (V2X) technology. In this study, we present a scheduling algorithm for the passing-through problem of the non-signalized intersection for vehicles with different priorities. First, we simplify the problem to an absolute value programming by dividing the intersection area into eight collision sections. Secondly, considering vehicles with the same priority and vehicles with different priorities scenarios, we proposed a rule-based scheduling algorithm to solve the problem by assigning a local optimal entering time for each vehicle in one scheduling step. Finally, we conducted extensive experiments to evaluate the performance of our proposed model and algorithm. The simulated results by SUMO showed that vehicles could pass the intersection efficiently without collision based on their priorities with the rule-based scheduling algorithm.**

Keywords—V2X, non-signalized intersection, rule-based algorithm, CAV, SUMO

I. Introduction

Nowadays, automated driving can liberate human beings from complicated and repeated driving tasks and make traffic more efficient[1]. Congestion or collision may occur when automated vehicles from different lanes merge into the intersection. So intersection scheduling has received considerable research attention in recent years [2-5]. Because most of the traditional intersections still rely on traffic signals, automated vehicles must leverage computer vision and artificial intelligence techniques to judge the traffic signal status and make passing strategies[6, 7]. Previous studies showed the signalized intersection has two main deficiencies as follows: (1) the complex environments (such as bad weather) may influence the computer vision and cause detection failures of the traffic signal status; (2) the traffic signal leads to the waste of a large amount of time as the number of automated vehicles is increasing [8, 9]. In this study, we take the non-signalized intersection as the research scenario to provide a smoother traffic flow and more efficient driving while also improving safety.

Connected and automated vehicles (CAVs) can provide driving safety, traffic efficiency, and road information in real-time via a novel technology called V2X by equipping with On Board Unit (OBU) [10, 11]. The emergence of CAVs has made it possible to implement efficient and collision-free scheduling for the non-signalized intersection [12]. CAVs in the non-signalized intersection area can transmit the information, including their position, speed, and acceleration (or deceleration) to the intersection scheduling center (ISC) through V2X communications. The ISC then can make passing strategies for vehicles with the received information. As the CAVs are becoming widely used, the more complex scenario has caught researchers' attention because different vehicles may have different urgency degrees (always called priority) to cross the intersection [13]. For example, ambulances will be assigned a high priority in the intersection area. Vehicles with high priority must pass the intersection without collision in prior.

With the above considerations, to address the passing-through problem in the non-signalized intersection, we need to deal with the following problems: (1) how to ensure vehicles pass through the intersection without collision and efficiently; (2) how to authorize vehicles with high priorities to pass through the intersection in prior. Focusing on this complex problem, we propose a scheduling algorithm that can assign an entering time for each arriving CAV with safety guarantee and priority principle.

The rest of this paper is arranged as follows. Section 2 introduces the traffic model, including the collision section and the passing-through time window in detail. In section 3, a rule-based scheduling algorithm is presented. We conduct the simulation and analyze the results in section 4. Finally, we conclude and direct our future work in section 5.

II. Model Of Problem

This study takes the most common non-signalized intersection shown in Fig 1 as the scenario to model. The intersection has four roads (E, S, W, and N). Each road has three lanes with different directions: go-straight, turn-left, and turn-right. We ignore the turn-right lane because vehicles in this lane do not affect vehicles in other lanes. The remaining eight lanes are marked with an ID. The *ith* lane is marked as L_i.

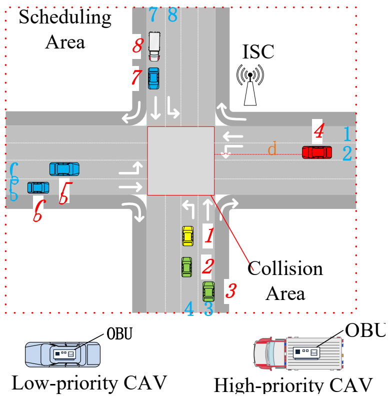


Fig. 1. Non-signalized intersection scenario

In this study, following the Ref. [14], we make three assumptions for this scenario:

(1) The intersection scheduling center (ISC) deployed in the intersection is equipped with the communication technique. The ISC can assign an entering time for each arriving CAV with reliable real-time performance.

(2) All the vehicles in the scenario are CAVs. CAVs should immediately send their status information (such as the position, speed, acceleration, and priority) to the ISC in each scheduling step. Vehicles will keep their lane and velocity until the ISC returns the entering time, and overtaking is not allowed in this study.

(3) Communication delay is not considered in this study.

As shown in Fig 1, we divide the intersection area into two areas: the Scheduling Area and the Collision Area. The Scheduling Area is the ISC's valid communication area, within which the ISC schedules a passing strategy with the received status information from vehicles. The Collision Area is the central area at the intersection, where vehicles may congest or collide.

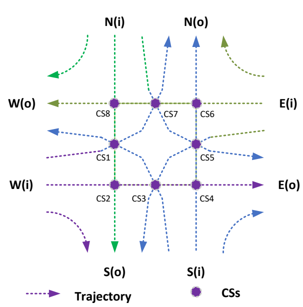


Fig. 2. CSs and possible conflict relations

In Fig 2, we divide the Collision Area into eight collision sections (CSs), from CS_1 to CS_8. CAV in each lane has a corresponding trajectory to pass through the Collision Area. For example, CAVs in L_4 would pass through the Collision Area with the south-to-west trajectory. We denote Ps_i as the go-straight lanes' passing CSs set and Pl_i as the turn-left lanes' passing CS set. The CS_i is in front of CS_j in passing CS set if a CAV passes through CS_i before CS_j on L_k. Each lane's passing CS set is shown in Table 1. We can observe that CAVs in L_3 may collide with CAVs in L_1 at CS_6. On the other hand, we can find those vehicles in the lanes L_3 and L_1 may collide at CS_6. The set $\{L_1, L_3\}$ is regarded as the conflict lane set of CS_6. When the ISC schedules the passing sequence of vehicles at CS_6, we can only consider vehicles in the conflict lanes to reduce the computation overhead.

TABLE I. THE MAPPING BETWEEN LANE ID AND CS SET

Lane ID	Ps_i	Lane ID	Pl_i
L_1	$\{CS_6, CS_7, CS_8\}$	L_2	$\{CS_5, CS_3\}$
L_3	$\{CS_4, CS_5, CS_6\}$	L_4	$\{CS_3, CS_1\}$
L_5	$\{CS_2, CS_3, CS_4\}$	L_6	$\{CS_1, CS_7\}$
L_7	$\{CS_8, CS_1, CS_2\}$	L_8	$\{CS_7, CS_5\}$

As shown in Fig 1, the width of each lane is w. We can get the position of each CS. For estimating the entering time of a CAV in each CS, we need to know the trajectory length: (1) for CAVs in go-straight lanes, the trajectory length is 4w; (2) for CAVs in turn-left lanes, we define that the trajectory length between the Collision Area boundaries and the CS is w_1 and the trajectory length between two CSs is w_2. The kth vehicle arrives at the Collision Area at the entering time t_i^k. Assuming that the velocity remains at v in Collision Area, the entering time set of each CS in Ps_i and Pl_i is shown in (1) and (2), respectively.

$$\{t_i^k + \frac{w}{2v}, t_i^k + \frac{2w}{v}, t_i^k + \frac{7w}{2v}\}, \quad (1)$$

$$\{t_i^k + \frac{w_1}{v}, t_i^k + \frac{w_1 + w_2}{v}\}, \quad (2)$$

We denote $\mathrm{PTW}_{i,k}$ as the passing-through time window of CS_i on the L_k. We denote that each CAV's length is d and the safety distance of vehicles is $\mathrm{d_s}$. Taking into account the specific trajectory of each lane, we could estimate their passing-through time window using their entering time of the Collision Area, such as $\mathrm{PTW}_{1,6} = (t_6^k + \frac{w_1}{v}, t_6^k + \frac{w_1}{v} + \frac{L}{v}]$. As one CS can only be occupied by one vehicle simultaneously, two passing-through time windows of the same CS cannot intersect. For example, $\mathrm{PTW}_{1,6} \cap \mathrm{PTW}_{1,4} = \left(t_6^i + \frac{w_1}{v}, t_6^i + \frac{w_1}{v} + \frac{L}{v}\right] \cap \left(t_4^j + \frac{w_1+w_2}{v}, t_4^j + \frac{w_1+w_2}{v} + \frac{L}{v}\right] = \emptyset$ is equivalent to $t_4^j + \frac{w_1+w_2}{v} - \left(t_6^i + \frac{w_1}{v} + \frac{L}{v}\right) \geq \frac{d_s}{v}$ or $t_6^i + \frac{w_1}{v} - (t_4^j + \frac{w_1+w_2}{v} + \frac{L}{v}) \geq \frac{d_s}{v}$. With the passing-through time window model, the collision avoidance constraints are formulated as a linear constraint.

III. RULES-BASED SCHEDULING ALGORITHMS

First-come-first-serve (FCFS) method [15] is a standard and classic scheduling method in current research. FCFS leads to a local optimal solution in many situations. In some simple scenarios, this method can solve the passing-through problem with less time complexity. But the method will fail in more complex scenarios, such as vehicles with different priorities. Thus, in this section, we propose the rule-based scheduling algorithm to solve the new passing-through problem.

A. The rules-based algorithm for CAVs with the same priority

We first investigate with the simplest scenario, where all the CAVs have the same priority. In this scenario, the arriving CAVs send their status information to the ISC at the beginning of each scheduling step. ISC has a container Q to store vehicle status information, which is first-in-first-out (FIFO). Suppose there are n_i CAVs on lane L_i at one scheduling step, thus the number of CAVs in Q is $n = \sum n_i$. At the scheduling step, ISC traverses Q and assigns an entering time for each CAV at each loop. Let $t = (t_1, t_2, \dots t_i, \dots t_n)^T$ represent the entering time of the Collision Area for CAVs. We set the object function for the CAV is to minimize the entering time of CAV_i.

CAVs can drive in the intersection area at speed in velocity range $[v_{min}, v_{max}]$. At each scheduling step, ISC can calculate the fastest arrival time $t_i^{min}$of the arriving CAV to arrive at the Collision Area as

$$t_i^{min} = t_i^0 + \frac{v_{max}-v_i^0}{a_{max}} + \frac{v_{max}-v}{a_{max}} + \frac{d-\frac{v_{max}^2-{v_i^0}^2}{2a_{max}}-\frac{v_{max}^2-v^2}{2a_{max}}}{v_{max}}, \quad (4)$$

where t_i^0 represents the arriving time of CAV_i, v_i^0 represents the arriving velocity of CAV_i, and v represents the velocity to pass through the Collision Area. v_{max} represents the maximum speed allowed in the Scheduling Area(vehicles with different priorities have different v_{max}). d represents the distance between the CAV and the Collision Area. Now we obtain the first constraint $\mathrm{t}_i \geq t_i^{min}$.

As overtaking is not allowed at the intersection area, the entering time on the same lane should be in increasing order. So the latest entering time on the lane L_i in last scheduling step is $t_i^{n_i}$. Let the t_h represent the safe time headway. We can obtain the second constraint $\mathrm{t}_i \geq t_i^{n_i} + t_h$ to ensure collision avoidance for the CAVs in their lanes.

The first vehicle in Q can be assigned an entering time, only considering the above two constraints. But ISC must consider the passing-through time window (PTW) constraints when it assigns entering time for other vehicles in Q. CAV_i $(i = 2,3, \dots n)$ must avoid collision with the first $\mathrm{i}-1$ vehicles in Q. As shown in Fig 1, CAV_7 on L_7 may collide with CAV_1, CAV_2, or CAV_6 at CS_1 and may collide with CAV_5 at CS_1. So the entering time of CS occupied by CAV_7 is $\{\mathrm{t}_7 + \frac{w}{2v}, \mathrm{t}_7 + \frac{2w}{v}, \mathrm{t}_7 + \frac{7w}{2v}\}$. We can get the PTW constraints of CAV_7 as

$$\begin{aligned}
&\left|\left(t_7 + \tfrac{2w}{v}\right) - \left(t_1 + \tfrac{w_1+w_2}{v} + \tfrac{L}{v}\right)\right| \geq \tfrac{d_s}{v},\\
&\left|\left(t_7 + \tfrac{2w}{v}\right) - \left(t_2 + \tfrac{w_1+w_2}{v} + \tfrac{L}{v}\right)\right| \geq \tfrac{d_s}{v},\\
&\left|\left(t_7 + \tfrac{7w}{2v}\right) - \left(t_6 + \tfrac{w}{2v} + \tfrac{L}{v}\right)\right| \geq \tfrac{d_s}{v},\\
&\left|\left(t_7 + \tfrac{2w}{v}\right) - \left(t_5 + \tfrac{w_1}{v} + \tfrac{L}{v}\right)\right| \geq \tfrac{d_s}{v}.
\end{aligned} \quad (5)$$

Correspondingly, we can transform the PTW constraints into matrix expressions $|T_7 + b_7| \geq c_7$, where

$$T_7 = \begin{pmatrix} t_7 \\ t_7 \\ t_7 \\ t_7 \end{pmatrix},$$

$$b_7 = \begin{pmatrix} -t_1 + \frac{2w}{v} - \frac{w_1+w_2}{v} - \frac{L}{v} \\ -t_2 + \frac{2w}{v} - \frac{w_1+w_2}{v} - \frac{L}{v} \\ -t_6 + \frac{2w}{v} - \frac{w}{2v} - \frac{L}{v} \\ -t_5 + \frac{7w}{2v} - \frac{w_1}{v} - \frac{L}{v} \end{pmatrix}, \; c_7 = \begin{pmatrix} \frac{d_s}{v} \\ \frac{d_s}{v} \\ \frac{d_s}{v} \\ \frac{d_s}{v} \end{pmatrix}. \quad (6)$$

Similarly, we derive all other matrix expressions $|T_i + b_i| \geq c_i, \mathrm{i} = 1,2, \dots, \mathrm{n}$, where the number of the rows of T_i, b_i and c_i is equal to the number of PTW constraints of CAV_i. Then the scheduling problem is transformed into

$$\begin{aligned}
&\min t_i ,\\
&s.t. \quad t_i \geq t_i^{min},\\
&\qquad\quad t_i \geq t_i^{n_i} + t_h,\\
&\qquad\quad |T_i + b_i| \geq c_i.
\end{aligned} \quad (6)$$

B. The rules-based algorithm for CAVs with different priorities

The problem will be more complicated when CAVs have different priorities. Now the rule-based mechanism's main target is to ensure that the high-priority vehicle can pass through the intersection in prior. The front low-priority CAVs often delay high-priority CAV if we only consider the priority of heading vehicles on each lane. We proposed the rules-based mechanism, which consists of the following two rules:

(1) Each lane is assigned a priority p_L, which is equal to the maximum vehicle priority p_v of the CAV queue q_i on that lane. ISC determines whether to update the lane priority according to the priorities of the arriving or the departing CAVs. In other words, lane priority keeps the maximum value of the CAV priority of the lane.

(2) In each scheduling step, CAVs on the highest-priority lane are assigned an entering time of *Collision Area* in prior. CAVs in other low-priority lanes should slow down to the low-priority velocity simultaneously to make way for the CAVs on high-priority lane.

We denote $Q_i\,(i = 1,2, \dots, 8)$ as the vehicle information container of L_i, e.g., $Q_7 = \{CAV_7, CAV_8\}$. In Fig 1, CAV_8 is an ambulance with the highest priority at the Scheduling Area, so L_7 is assigned the highest priority based on the rule (1). After the ISC assigns the entering time for CAV_7 and CAV_8, ISC will assign entering time for the remaining vehicles based on the simple rule-based algorithm. Similarly, we have the scheduling problem as

$$\begin{aligned} \min t_i \quad &, \\ s.t. \quad t_i &\geq t_i^{min}, \\ t_i &\geq t_i^{n_i} + t_h, \\ |T_i + b_i| &\geq c_i. \end{aligned} \tag{7}$$

Let $b_i[j]$ and $c_i[j]$ be the j^{th} component of b_i and c_i. So $|t_i + b_i[j]| \geq c_i[j]$ can be transformed into $t_i \geq c_i[j] - b_i[j]$ or $t_i \leq -c_i[j] - b_i[j]$. We denote $t_i^0 = \max\{t_i^{min}, t_i^{n_i} + t_h\}$. We can get the first sub-problem of the problem (7) as

$$\begin{aligned} \min t_i \quad &, \\ s.t. \quad t_i &\geq t_i^0, \\ |t_i + b_i[1]| &\geq c_i[1]. \end{aligned} \tag{8}$$

By employing the OET algorithm[20], the optimal solution of the sub-problem (8) can easily be solved as

$$t_i^1 = \begin{cases} c_i[1] - b_i[1], & if - c_i[j] - b_i[1] < t_i^0 < c_i[1] - b_i[1] \\ t_i^0, & otherwise. \end{cases} \tag{9}$$

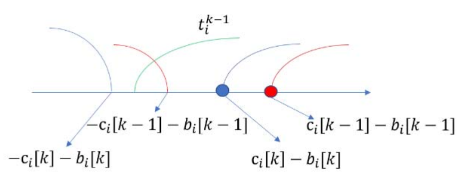


Fig. 3. The scheduling step of the kth sub-problem

Next, we add the constraints in $|T_i + b_i| \geq c_i$ to the sub-problems in turns to obtain new sub-problems and solve it. If we only consider the new time window constraints, the situation in the following figure may appear. The blue point in the figure is the optimal solution. Obviously this conflicts with the time window constraint in the previous round, so we also need to consider the time window constraint in the previous round of scheduling, so that we can get a feasible optimal solution. For example, we add the *kth* $|t_i + b_i[k]| \geq c_i[k]$ *PTW* constraint into the *k-1th* problem (9), i.e.,

$$\begin{aligned} \min t_i \quad &, \\ s.t. \quad t_i &\geq t_i^{k-1}, \\ |t_i + b_i[k-1]| &\geq c_i[k-1], \\ |t_i + b_i[k]| &\geq c_i[k]. \end{aligned} \tag{10}$$

We can get the optimal solution t_i^k of sub-problem (10) combined with the optimal solution t_i^{k-1} of the *k-1th* sub-problem. We obtain the current optimal solution through the intersection of the intervals on the number axis. Finally, we can get the optimal solution to the problem (7) by adding all the constraints in $|T_i + b_i| \geq c_i$ to sub-problem (8).

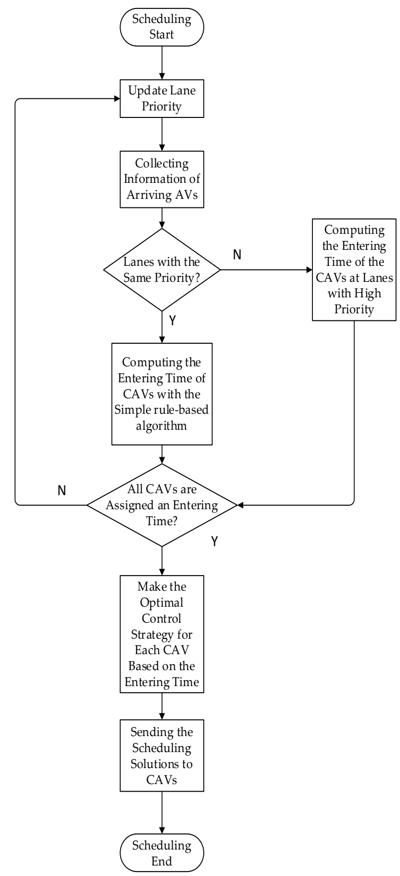


Fig. 4. The flow chart of the rules-based algorithm

IV. Simulation and Experiment

This section gave out scenarios and parameters setup and conducted simulations to verify the proposed models and the rule-based algorithms.

A. Simulation setup

We used SUMO to generate the simulation scenario in this section. SUMO allows us to handle large, complicated road networks at a microscopic (vehicle-level) scale [16]. The simulation allows us to address a broad set of traffic management topics. Traffic Control Interface(TraCI) for Python gives access to a running road traffic simulation, and it allows us to retrieve values of simulated objects and to manipulate their behavior "on-line"[17]. Fig 4 illustrates the snapshot of the four-bidirectional six-lane road scenario generated by SUMO, where the Scheduling Area's length is set as 300m, and the width of each lane is 4m.

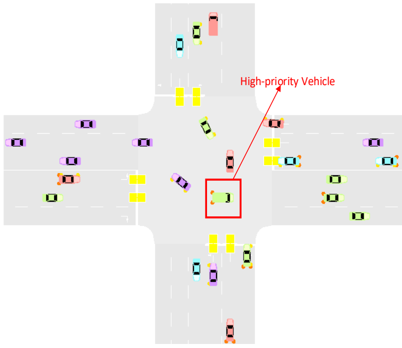


Fig. 5. The simulation scenario generated by SUMO

To verify this algorithm's applicability in different traffic scenarios, we generate traffic flow in two different ways, i.e., all the CAVs had the same priority in a *scheduling step* (*scenario 1*), and CAVs had different priorities in a *scheduling step* (*scenario 2*). We set vehicles to enter from four directions (0 for the east, 1 for the south, 2 for the west and 3 for the north) respectively, and the destination direction of each vehicle was randomly allocated from go-straight, turn-left and turn-right to ensure the randomness of traffic flow in each lane. In *scenario 1*, all the CAVs drive at the velocity limit of 20m/s in the Scheduling Area. For *scenario 2*, we set the vehicles to have only two priorities: high priority and low priority. According to the different distribution of high-priority vehicles in one *Scheduling Step*, we get four different cases in Table 3. For example, Case 1 represented that one high-priority vehicle occurred at the queue head on one lane. CAVs with low priority should drive at *Low Priority Velocity* to make way for the high-priority CAVs. All the CAVs in both scenarios must adjust velocity to 10m/s before entering the Collision Area. The parameter settings are summarized in Table 2.

We chose traditional traffic light and the ad hoc negotiation-based algorithm (following FCFS order) as benchmarks to compare with our rules-based scheduling algorithm.

TABLE II. PARAMETERS SETTING

Parameters	Value	Parameters	Value
Min Velocity	0 m/s	Low Priority Velocity	15 m/s
Max Velocity in *Scheduling Area*	20 m/s	Safe distance	5 m
Velocity in *Collision Area*	10 m/s	Initial Speed	15 m/s
Max Ac/Deceleration	5/-5 m/s²	Length of *Scheduling Area*	300 m
Scheduling Step	5s	Width of Lanes	4m

TABLE III. FOUR CASES FOR SCENARIO 2

	High-priority vehicle occurred on one lane	High-priority vehicles occurred on multiple lanes
At the queue head	Case 1	Case 2
In the middle of the queue	Case 3	Case 4

B. Evaluation metrics

We considered the following three metrics to evaluate the performance of our proposed algorithm:

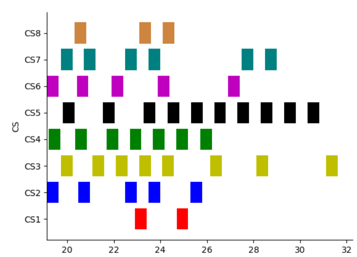
1) PTW: It can convert competition problems from the spatial domain to the spatiotemporal domain. Therefore, no overlap between PTWs is allowed to ensure that CAVs pass the Collision Area without collision;

2) Vehicles Sequence: It refers to the order in which vehicles enter the conflict zone. Vehicles Sequence can intuitively indicate whether high-priority vehicles pass through the Collision Area efficiently;

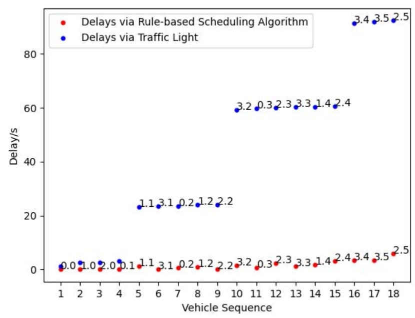
3) Delay: It refers to the delayed time caused by the intersection, which can be evaluated by the time gap between the Collision Area's assigned entering time and the fastest arrival time.

C. Experiment results analysis

In Scenario 1, the rule-based algorithm degraded to the FCFS. In Fig 5, we had the following two observations: (1) no overlap between the PTWs existed in the same CS, that is, vehicles entering the intersection area in the current Scheduling Step can pass through the Collision Area without collision; (2) compared with traffic signal scheduling, the rule-based scheduling algorithm can significantly reduce vehicle delays. In general, the rule-based algorithm can ensure vehicles pass through the intersection without collision and efficiently in scenario 1.



(a)



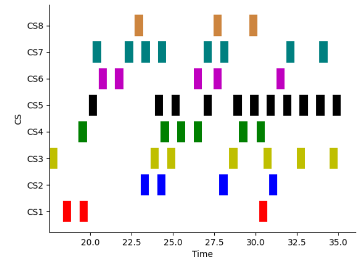
(**b**)

Fig. 6. Figs for Case 1. (a) PTWs of eight CSs for Case 1; (b) Vehicle delays for Case 1.

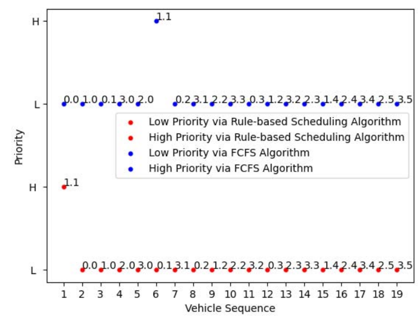
We simulate four cases respectively, and the high-priority vehicle information of each case is shown in Table IV. Fig 6 - 9 showed the PTWs, vehicle sequence and vehicle delays for the four cases. In Case 1 and Case 2, we found that high-priority vehicles would enter the Collision Area first under the rules-based scheduling algorithm, and the sequence of the low-priority vehicles under the rules-based scheduling algorithm was equal to that of FCFS. High-priority vehicle delays are significantly reduced, e.g., in Case 1 the delay of high-priority vehicle 1.1 was reduced from 1.77s to 0s through the rules-based scheduling algorithm, and the delay time of other low-priority vehicles increased. When the high-priority vehicles occurred at the lanes' middle position, the lanes were assigned a high priority. Vehicles on high-priority lanes are assigned Entering Time earlier according to the rules-based scheduling algorithm. For example, vehicle 3.1 is a high-priority vehicle in case 3, lane L_7 was assigned a high priority. Fig 9 showed that both vehicle 3.0 and vehicle 3.1 on lane L_7 would enter the Collision Area in prior. The delay of the high-priority vehicle via the rules-based scheduling algorithm was reduced to 0s from 2s via the FCFS algorithm. Similarly, the delay of other low-priority vehicles has increased, but it is also within the tolerable range. In general, we observed that the rule-based algorithm could ensure that the vehicle passes through the Collision Area efficiently and without collision under four cases.

TABLE IV. HIGH-PRIORITY VEHICLE INFORMATION IN FOUR CASES

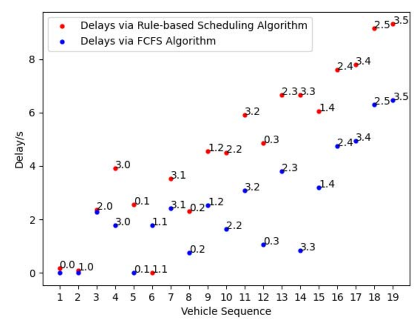
Cases	High-priority Vehicle ID	Destination Direction
Case 1	1.1	Turn-left
Case 2	0.1, 1.2	Go-straight, Go-straight
Case 3	3.1	Go-straight
Case 4	2.2, 0.2	Go-straight, Turn-left



(a)

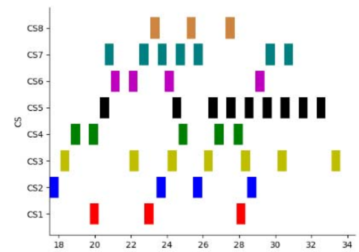


(b)

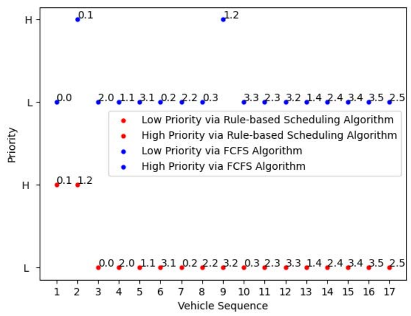


(c)

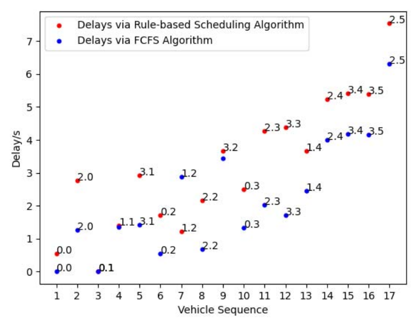
Fig. 7. Figs for Case 1. (a) PTWs of eight CSs for Case 1; (b) Vehicle Sequence for Case 1; (c) Vehicle delays for Case 1.



(a)

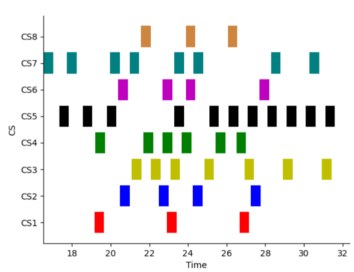


(b)

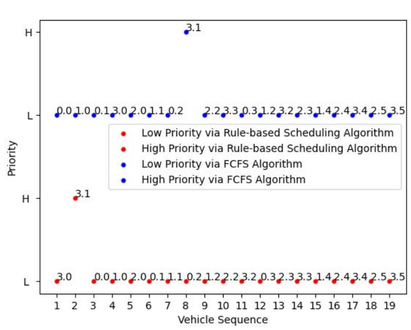


(c)

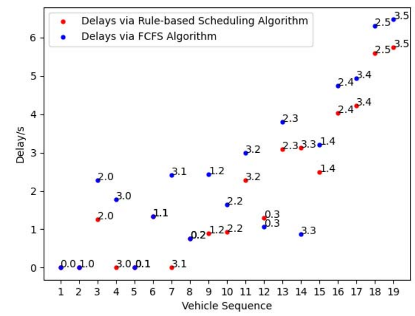
Fig. 8. Figs for Case 2. (a) PTWs of eight CSs for Case 2; (b) Vehicle Sequence for Case 2; (c) Vehicle delays for Case 2.



(a)

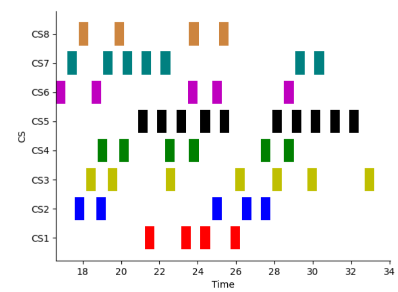


(b)

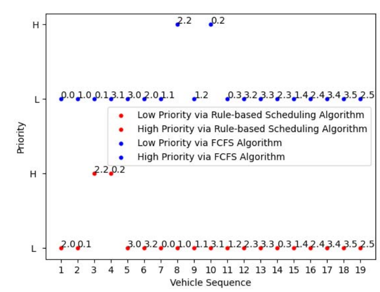


(c)

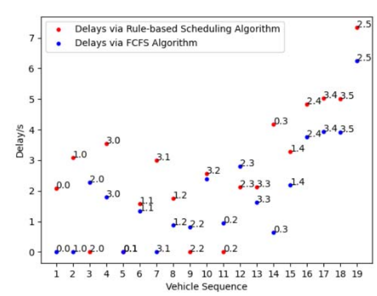
Fig. 9. Figs for Case 3. (a) PTWs of eight CSs for Case 3; (b) Vehicle Sequence for Case 3; (c) Vehicle delays for Case 3.



(a)



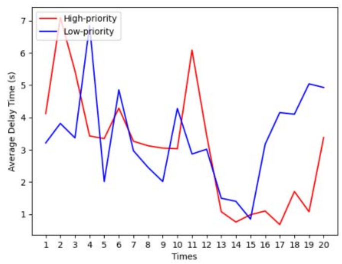
(b)



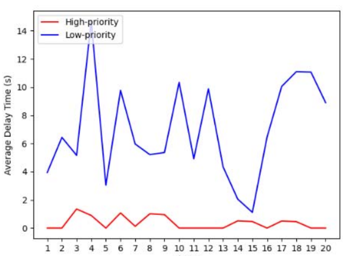
(c)

Fig. 10. Figs for Case 4. (a) PTWs of eight CSs for Case 4; (b) Vehicle Sequence for Case 4; (c) Vehicle delays for Case 4.

Finally, we conducted a total of 20 different sets of simulations for the above four cases. We calculated the average delay of high-priority vehicles and low-priority vehicles in each simulation. Fig 12 showed the experiment results for FCFS and the rules-based scheduling algorithm. By comparing Fig 10 (a) with Fig 10 (b), we found that the rule-based scheduling algorithm could guarantee a decrease in the delay of high-priority CAVs, and other low-priority vehicles were delayed a little.



(**a**)



(**b**)

Fig. 11. The average delay time of FCFS and the rule-based scheduling algorithm. (a) FCFS algorithm; (b) the rule-based scheduling algorithm.

Different results had shown that the proposed model and the algorithm had a good performance to schedule high-priority vehicles in prior and improved traffic efficiency.

V. Conclusion

In this study, we investigated the scheduling of CAVs at a non-signalized intersection. Focusing on this topic, we studied and proposed a passing-through time window model and the rule-based scheduling algorithm. Instead of assigning an entering time for each CAV to enter the Collision Area based on FCFS, the rule-based scheduling algorithm can schedule according to different vehicle priorities. The proposed model and algorithm are simulated in SUMO and Python. Via performing simulations in different scenarios, we conclude that CAVs are scheduled to pass the non-signalized intersection efficiently and without collision and high-priority CAVs in prior. Our future work will study the application of 5G technology for scheduling CAVs at the non-signalized intersection.

VI. Acknowledgement

I would like to express my great appreciation to my supervisor, Professor Shi for his valuable help and suggestions in the process of planning, writing, and revising this study. This research was funded by the China National Key Research and Development Program (NO.2018YFE0197700). Thanks for the support of the fund.

References

[1] Qian, B., et al., Toward Collision-Free and Efficient Coordination for Automated Vehicles at Unsignalized Intersection. IEEE Internet of Things Journal, 2019. 6(6): p. 10408-10420.

[2] Shen, X., et al., Cooperative Comfortable-Driving at Signalized Intersections for Connected and Automated Vehicles. IEEE Robotics and Automation Letters, 2020. 5(4): p. 6247-6254.

[3] Pourmehrab, M., et al., Optimizing Signalized Intersections Performance Under Conventional and Automated Vehicles Traffic. IEEE Transactions on Intelligent Transportation Systems, 2020. 21(7): p. 2864-2873.

[4] Chen, X., et al., Non-Signalized Intersection Network Management With Connected and Automated Vehicles. IEEE Access, 2020. 8: p. 122065-122077.

[5] Mirheli, A., et al., A consensus-based distributed trajectory control in a signal-free intersection. Transportation Research Part C: Emerging Technologies, 2019. 100: p. 161-176.

[6] Zhang, J., et al., Data-Driven Intelligent Transportation Systems: A Survey. IEEE Transactions on Intelligent Transportation Systems, 2011. 12(4): p. 1624-1639.

[7] Nazini, H. and V. Hemakumar, Optimized traffic signal control system at traffic intersections using VANET. 2013: p. 305-312.

[8] Bian, Y., et al., Cooperation of Multiple Connected Vehicles at Unsignalized Intersections: Distributed Observation, Optimization, and Control. IEEE Transactions on Industrial Electronics, 2020. 67(12): p. 10744-10754.

[9] Zhang, Y.J., A.A. Malikopoulos and C.G. Cassandras. Optimal control and coordination of connected and automated vehicles at urban traffic intersections. 2016: American Automatic Control Council (AACC).

[10] Rios-Torres, J. and A.A. Malikopoulos, A Survey on the Coordination of Connected and Automated Vehicles at Intersections and Merging at Highway On-Ramps. IEEE Transactions on Intelligent Transportation Systems, 2017. 18(5): p. 1066-1077.

[11] Edwards, T. Connected and automated vehicles: Concepts of V2x communications and cooperative driving. in Autonomous Passenger Vehicles. 2017.

[12] Chen, L. and C. Englund, Cooperative Intersection Management: A Survey. IEEE Transactions on Intelligent Transportation Systems, 2016. 17(2): p. 570-586.

[13] Zhang, K., et al., State-Driven Priority Scheduling Mechanisms for Driverless Vehicles Approaching Intersections. IEEE Transactions on Intelligent Transportation Systems, 2015. 16(5): p. 2487-2500.

[14] Qian, B., et al., Toward Collision-Free and Efficient Coordination for Automated Vehicles at Unsignalized Intersection. IEEE Internet of Things Journal, 2019. 6(6): p. 10408-10420.

[15] Meng, Y., et al., Analysis of Cooperative Driving Strategies for Nonsignalized Intersections. IEEE Transactions on Vehicular Technology, 2018. 67(4): p. 2900-2911.

[16] Mon, E.E., et al., Bottleneck Based Gridlock Prediction in an Urban Road Network Using Long Short-Term Memory. Electronics, 2020. 9(9): p. 1412.

[17] Lopez, P.A., et al., Microscopic Traffic Simulation using SUMO, in 2018 21st International Conference on Intelligent Transportation Systems (ITSC) 2018 21st International Conference on Intelligent Transportation Systems (ITSC). 2018. p. 2575-2582.