

Research Article

Distributed Cooperative Driving Strategy for Connected Automated Vehicles at Unsignalized Intersections Based on Monte Carlo Method

Haoming Li^(b),¹ Wei Dong,² Linjun Lu^(b),¹ Ying Wang^(b),¹ and Xianing Wang^(b)

¹Department of Transportation Engineering, School of Naval Architecture, Ocean and Civil Engineering, Shanghai Jiao Tong University, Shanghai 200240, China ²Department of Engineering Technical, China FAW Group Co., Ltd., Changchun 130000, China

Correspondence should be addressed to Linjun Lu; linjunlu@sjtu.edu.cn

Received 18 October 2023; Revised 18 November 2023; Accepted 7 December 2023; Published 6 March 2024

Academic Editor: Peng Hang

Copyright © 2024 Haoming Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

One of the most important goals of cooperative driving is to control connected automated vehicles (CAVs) passing through conflict areas safely and efficiently without traffic signals. As a typical application scenario, allocating right-of-way reasonably at unsignalized intersections can effectively avoid collisions and reduce traffic delays. Proposed here is a new cooperative driving strategy for CAVs at unsignalized intersections based on distributed Monte Carlo tree search (MCTS). A task-area partition framework is also proposed to decompose the mission of cooperative driving into three main tasks: vehicle information sharing, passing order optimization, and trajectory control. Based on the schedule tree of the vehicle passing order, the root parallelization of MCTS combined with the majority voting rule is used to explore as many feasible passing orders (leaf nodes) as possible in a distributed way and find a nearly global-optimal passing order within the limited planning time. The aim is for CAVs to perform proper trajectory adjustments based on the obtained passing order to minimize traffic delays while making the slightest acceleration adjustments. A coupled simulation platform integrating SUMO and Python is developed to construct the unsignalized intersection scenarios and generate the proposed strategy significantly enhances efficiency, safety, comfort, and emission, aligning well with innovative and environmentally friendly urban mobility aspirations.

1. Introduction

Connected automated vehicles (CAVs) are vital components of the new generation of transportation systems [1-3], and CAV-based traffic control is an effective way to improve safety, efficiency, and energy consumption [4]. With the help of V2X technology, CAVs can share their real-time operational data and communicate with roadside infrastructure to better coordinate their overall movement in the intelligent connected environment [5, 6].

The optimization of trajectories for CAVs has been recognized as a practical approach to enhance the overall efficiency of urban traffic systems [7]. Over the past decade, extensive research has been conducted on trajectory control for CAVs. A variety of control strategies have been

developed, including adaptive cruise control (ACC) [8], cooperative adaptive cruise control (CACC) [9, 10], model predictive control (MPC) [11–13], and deep reinforcement learning (DRL) control [14–17], are developed to optimize the trajectories of CAVs.

Intersections are the main bottlenecks for urban traffic [18], and congestion there causes great socioeconomic losses and increases travel delays significantly [19, 20]. As an indispensable part of traffic control, intersection management will change from traditional traffic-light control to unsignalized autonomous intersection management (AIM) for better coordination [21–23]. The main task of AIM is to control CAVs cooperatively to pass through the conflict areas of intersections safely and efficiently [24]. In recent years, researchers have found that the most critical factor of

cooperative driving at unsignalized intersections is the passing order of CAVs [25, 26], and there are two main types of cooperative driving strategies to determine the passing order: reservation-based and planning-based [24].

Reservation-based strategies use some heuristic rules to allocate the right-of-way for CAVs in a short period [27]. Dresner and Stone proposed an AIM strategy that allocates space resources to vehicles on a first-come, first-served (FCFS) basis [28, 29]. Choi et al. extended reservationbased cooperative control to multilane intersections [30]. Malikopoulos et al. proposed an optimal decentralized energy control framework for CAVs [31]. Zhang and Cassandras extended the framework further to include all possible turns and considered a joint energy-time optimal solution [32]. However, reservation-based strategies mainly follow the FCFS approach, and their performance is not good enough in many cases [24].

Planning-based strategies aim to find a globally optimal solution for CAVs by enumerating all possible passing orders [27], and most scholars formulate the problem as a mixed-integer linear programming optimization problem for minimizing the total traffic delay of the intersection [33, 34]. Li and Wang proposed the tree search method, the equivalent goal of which is to find the leaf node (passing order) corresponding to the optimal solution [25]. Xu et al. proposed a Monte Carlo tree search (MCTS)-based strategy to find a well-performing passing order within a limited planning time [35]. Zhang and Cassandras designed a dynamic resequencing scheme to optimize the passing order [36]. Apart from these methods, graph theory has been employed to determine the optimal passing order for multiple CAVs [37, 38]. However, a significant portion of the existing literature concentrates primarily on the feasibility of a conflict-free passing order solution or on deriving an optimal passing order through specific methods, often overlooking the aspect of computational complexity. With more vehicles, fewer passing orders are explored within the planning time, which brings difficulties for practical applications [39-41].

To address the above problem, we propose a new distributed cooperative driving strategy to maintain a good balance between performance and computation. The key idea is to utilize the constrained planning time to investigate nodes that have the potential to yield the optimal solution. To this end, the MCTS algorithm incorporating heuristic rules is used to accelerate the search process, and the root parallelization of MCTS combined with the majority voting rule is applied to implement the distributed cooperation and explore more leaf nodes. We also present a task-area partition framework for task decomposition matched with the strategy.

Note that Xu et al. [35] pioneered a centralized MCTSbased cooperative driving strategy at unsignalized intersections in which a roadside controller gathers information from all incoming CAVs to calculate a wellperforming passing order. Inspired by that work, this paper aims to elucidate further how to explore and evaluate more passing orders in a distributed way, thereby augmenting the solution's effectiveness. The main contribution is introducing a distributed cooperative strategy, which integrates root parallelization MCTS and the majority voting rule, into the proposed driving taskarea partition framework to determine a nearly globaloptimal passing order within the constraints of limited planning time.

The paper is organized as follows: Section 2 details the problem, Section 3 presents the new strategy, Section 4 introduces the details of simulation implementations, and Section 5 validates the effectiveness of the proposed strategy via the results of simulation experiments. Finally, Section 6 concludes the paper.

2. Problem Statement

The unsignalized six-lane intersection shown in Figure 1 involves three key concepts: (i) intersection physical area (IPA), (ii) control area (CA), and (iii) communication range (CR). The area within the circle of radius R_2 is the IPA in which collisions might happen, CR is within the circle of radius R_1 , and CA is within the CR but outside the IPA. The communication range is a logical concept wherein vehicles function as independent agents that communicate with each other and control themselves in real time. The intersection can be occupied simultaneously by multiple vehicles for better travel efficiency.

To simplify the problem, we make the following assumptions:

- (i) All vehicles are CAVs equipped with V2X communication devices and can share their real-time operational data (position, velocity, etc.)
- (ii) Lane changing is prohibited after entering the control area to ensure vehicle safety
- (iii) There are no communication time delays or package losses
- (iv) Vehicles move at a constant speed when passing through the intersection physical area

Because all vehicles have satisfactory lane-keeping ability, we focus only on longitudinal vehicle control. As shown in Figure 2, the innermost (leftmost) lane in the entrance direction allows vehicles to turn left or go straight, the middle lane is for going straight only, and the outermost (rightmost) lane allows vehicles to turn right or go straight.

The complexity of the unsignalized intersection control stems from the conflicting natures of various traffic movements, which typically have three conflicting modes: crossing, converging, and diverging. The three modes delineate the conflict relationships among traffic movements at intersection areas, exit lanes, and entrance lanes to prevent conflicting vehicles from passing through intersections at the same time. Figure 2 also shows the spatial distribution of the conflict points. According to the geometry of the intersection, the conflict points are divided into 64 crossing ones, eight converging ones, and eight diverging ones.

After entering CA, each CAV is treated as an independent agent. CAVs calculate the passing orders based on the collected driving data, respectively, and the final



FIGURE 1: A typical unsignalized intersection scenario.



FIGURE 2: Spatial distribution of conflict points.

uniform passing order is decided by all agents from the above calculated passing orders. Then, agents perform corresponding trajectory planning and adjustments to avoid collisions in IPA. The distributed cooperative strategy aims to minimize traffic delays while making the slightest acceleration adjustments at the unsignalized intersection. So, we have the following evaluation function:

$$D = \sum_{i \in S} \left(t_{\text{actual},i} - t_{\min,i} \right), \tag{1}$$

$$A = \sum_{i \in S} a_i, \tag{2}$$

$$J = \alpha_1 \cdot D + \alpha_2 \cdot A, \tag{3}$$

where *i* is the *i*th CAV entering CA, *S* is the set of all CAVs in CA currently, $t_{\min,i}$ represents the travel time spent by CAV_i from entering CA to passing through IPA at the maximum speed, $t_{actual,i}$ denotes the actual travel time of CAV_i, *D* is the intersection's total traffic delay, a_i represents the number of acceleration adjustments required for CAV_i to pass through the intersection safely (calculated for each 1 m/s² change in acceleration), *A* is the total number of required acceleration adjustments for all CAVs, and α_1 and α_2 are the weighing coefficients.

Recognizing that the efficiency of traffic flow at intersections is predominantly influenced by the passing order of the CAVs, we employ the schedule tree theory and frame the entire issue as a tree search problem, wherein each leaf node signifies a distinct passing order [24, 25]. We take the simple intersection scenario shown in Figure 3 as an example. The passing order ABCD indicates the priorities of the four CAVs. If two CAVs have the same spatiotemporal conflict point, such as CAV_b and CAV_c , then the one ranking lower (CAV_c) in the passing order adjusts its trajectory by slowing down to reach the conflict point later than expected to avoid collision. However, two CAVs without conflict, such as CAV_a and CAV_b, can pass through the intersection simultaneously. To calculate all acceleration adjustments required to avoid potential collisions, it is necessary to ensure that the lower-priority CAV takes account of any adjustments made by the higher-priority CAV.

Figure 4 shows a schematic of building the schedule tree for the scenario shown in Figure 3. All possible passing orders are generated as leaf nodes in the bottom layer of the schedule tree.

If a passing order is given, then the total traffic delay and the required collision-avoidance acceleration adjustments for all CAVs in the passing order can be derived directly from Algorithm 1.

In the requirement function, CAV_i performs the conflict analysis judgment in turn with CAVs with higher priority in the passing order based on Figure 2. If there is a conflicting trajectory between the two CAVs and the expected time interval to reach the conflict point is within the given threshold [35], CAV_i makes the collision-avoidance acceleration adjustments, and the requirement function returns the Boolean value true. The CAVs needing acceleration adjustments update their trajectories after calculating the required adjustments to ensure safe trajectory adjustments for subsequent CAVs in the passing order. The time complexity of Algorithm 1 is O(n).

3. Methodology

This section proposes a task-area partition framework for cooperative driving task decomposition. Moreover, it presents a root parallelization MCTS method with the majority voting rule, implementing the distributed cooperation while exploring and evaluating more passing orders to accomplish the driving tasks.

To clearly articulate the differences between the proposed distributed MCTS-based cooperative driving strategy (D-MCTS) and the existing classical centralized MCTS-



FIGURE 3: An intersection scenario with four vehicles.



FIGURE 4: The schedule tree stemmed from the intersection scenario shown in Figure 3.

based cooperative driving strategy (C-MCTS), Figure 5 demonstrates the methodological framework of the two MCTS-based cooperative driving strategies.

3.1. Task-Area Partition Framework. The proposed task-area partition framework decomposes the mission of cooperative driving into three main tasks: (i) vehicle information sharing, (ii) passing order optimization, and (iii) trajectory control. In Figure 1, the intersection functional area (communication range) is partitioned accordingly into four areas: observation area (OBA), optimization area (OPA), execution area (EXA), and intersection physical area (IPA), the ranges of which are d_{oba} , d_{opa} , d_{exa} , and R_2 , respectively. In each area, CAVs are assigned the following different tasks:

- First, in OBA, approaching CAVs share their realtime operational data (position coordinates, speed, acceleration, current lane, target lane, etc.) based on the V2X information interaction technology.
- (2) Then, in OPA, root parallelization is applied to implement the distributed cooperation (i.e., each CAV calculates a nearly global-optimal passing order

Input: A passing order P

Output: The total acceleration adjustments A of the covered CAVs and their required acceleration adjustments a_i, respectively

- (1) Initialize a_i as 0 (2) for each $i \in [1, \text{ length } (P)]$ do
- (3) $t_i = \text{actual_time}(i) - \min_\text{time}(i)$ [35]
- (4)adjustment_required = Requirement (i)
- The Requirement function determines whether CAV, needs to make the acceleration adjustment (5)
- while adjustment_required do (6)
- (7) $a_i = \text{acc_calculate}(i)$
- (8)for each $j \in [i, \text{ length } (P)]$ do (9) if $lane_i = = lane_i$ then (10) $a_i + = a_i$ end if (11)end for (12)(13)adjustment_required = Requirement (i) (14)end while
- (15) end for (16) $D = \sum_{i=1}^{\text{length}(P)} t_i, A = \sum_{i=1}^{\text{length}(P)} a_i,$ (17) $J = \alpha_1 \cdot D + \alpha_2 \cdot A$

ALGORITHM 1: Passing order to trajectory interpretation.



FIGURE 5: Methodological framework of the two MCTS-based cooperative driving strategies. (a) Centralized MCTS-based cooperative driving strategy. (b) Distributed MCTS-based cooperative driving strategy.

based on the MCTS algorithm with heuristic rules, and then, all CAVs apply the majority voting rule to determine the final uniform passing order) to specify the following driving behaviors of all CAVs.

- (3) Next, in EXA, each CAV carries out the corresponding trajectory planning and adjustments in real-time to meet the desired driving trajectory determined in task 2 and keeps intervehicle safety gaps to arrive at IPA on time.
- (4) Finally, in IPA, the driving behaviors of CAVs are locked, and no further trajectory adjustments are made. CAVs pass through the intersection (then become departing vehicles) and leave the intersection area safely.

This framework provides a new solution for designing multivehicle cooperative driving strategies by assigning sensing, decision, and control tasks to different task areas.

3.2. MCTS-Based Cooperative Driving Strategy. Herein, we apply MCTS combined with heuristic rules to select leaf nodes that have the potential to be the corresponding optimal passing order [35]. In MCTS, each node in the search tree is assigned a score equal to equation (3) of its corresponding passing order to evaluate its potential, and the MCTS algorithm uses these scores to determine which branch of the tree should be explored.

MCTS establishes a search tree iteratively. Taking the scenario in Figure 3 as an example, Figure 6 shows one iteration of the MCTS-based strategy, which includes four steps: selection, expansion, simulation, and backpropagation [42].

3.2.1. Selection. We start at the root node and pick the highest-scoring child node recursively until reaching either the most urgent expandable node or a terminal state. The score for traversing the tree in MCTS is defined as the following tree policy and is given by [43]:

$$\arg\max_{k}G_{k} + C\sqrt{\frac{\ln\left(t\right)}{t_{k}}} \tag{4}$$

where G_k is the score of child node k, C is a weighting parameter, t is the number of times the currently searched node was visited, and t_k is the number of times that child node k was visited. An expandable node refers to one that is not a leaf node but has unvisited child nodes. Equation (4) is an attempt to balance exploration and exploitation.

3.2.2. Expansion. We randomly select an unvisited child node of the selected most urgent expandable node as the new node to add to the tree unless the selected node is at a terminal state.

3.2.3. Simulation. The simulation policy is used to directly obtain a leaf node (complete passing order) based on the current new searched node (partial passing order) to evaluate its potential. Because the complete passing order

generated by random sampling cannot help us to effectively evaluate the true potential of the current new searched node during the simulation, we add the following two heuristic rules to the simulation process for deciding which node (CAV) should be expanded:

- (i) For CAVs in the same lane, add the current leading CAV first.
- (ii) For CAVs passing through the same conflict point, add the one with the less desired arrival time first.

We update the score of the current new searched node after simulation via the following four steps:

- (a) Calculate the weighted summation of total delay and acceleration adjustments \overline{j}_k of the partial passing order corresponding to the current new searched node.
- (b) Calculate the weighted summation of total delay and acceleration adjustments \hat{j}_k of the complete passing order corresponding to the best leaf node of the current new searched node via simulation.
- (c) Normalize \overline{j}_k and \hat{j}_k into [0,1] using

$$q_{k} = 1 - \frac{\left(j_{k} - j_{k,\min}\right)}{\left(j_{k,\max} - j_{k,\min}\right)},$$
(5)

where $j_{k, \text{max}}$ and $j_{k, \text{min}}$ are the maximum and minimum weighted summation of total delay and acceleration adjustments among the sibling nodes of node k, respectively.

(d) Calculate the score G_k of the current new node as

$$G_k = \omega \overline{q}_k + (1 - \omega) \widehat{q}_k \tag{6}$$

where ω is a weighting parameter.

3.2.4. Backpropagation. The result of the simulation is backpropagated through the selected current new searched nodes to the root node for updating the scores of all parent nodes.

During the establishment of the search tree, the current optimal passing order is updated dynamically and continuously. Once the computation budget is reached, the MCTS terminates and returns the current-optimal passing order. The planned total traffic delay and required acceleration adjustments of the CAVs are determined using Algorithm 1, and the simulation process of MCTS is given by Algorithm 2.

3.3. Distributed Cooperative Driving. Distributed cooperative driving at an unsignalized intersection can be achieved by running simulations simultaneously via multiple agents in parallel (MCTS parallelization), which allows the whole multiagent system to run more MCTS simulations, i.e., explore as many leaf nodes (passing orders) as possible to evaluate their potential within a limited computation budget [43]. There are three main types of parallelization methods for MCTS: leaf, root, and tree [44].

Journal of Advanced Transportation



FIGURE 6: One iteration of the MCTS-based cooperative driving strategy.

Input: Operational data of all CAVs **Output:** A possible passing order

- (1) Choose the uncovered leading CAV of each lane as the candidate CAVs and calculate their arrival times to all conflict points on their desired trajectories.
- (2) Add the CAV whose arrival times to all conflict points (compared to other candidate CAVs) are least into the passing order. If not, randomly select one.
- (3) Repeat steps 1 and 2 until a complete passing order is generated.
- (4) The objective value (3) of the passing order \hat{j}_k can be derived by Algorithm 1.



Leaf parallelization is applied to improve simulation results and can be implemented in a distributed strategy. This method requires one agent to establish the search tree, while the other agents participate in the parallel simulations, which brings the problem of choosing a CAV to build the search tree. Root parallelization and tree parallelization provide a multiagent method in which each CAV can contribute to the overall strategy as an independent agent. In root parallelization, multiple independent trees are built by separate CAVs with no information communicated before unifying the results [45]. Tree parallelization brings the problems of maintaining a tree among all CAVs and selecting one CAV as the tree holder.

We decided to use root parallelization, following the example of Kurzer et al., who used root parallelization MCTS successfully in cooperative multiagent system trajectory planning for automated vehicles [46]. Each CAV acts as an independent agent in root parallelization to calculate its own MCTS solution.

Soejima et al. explored root parallelization in the computerized Go field; they compared the strategy based on the majority voting rule versus the average voting rule and found that the former was superior [47]. Thus, we use the following majority vote rule to unify the solutions calculated by all CAVs. Once a CAV determines a passing order, it votes for that passing order and shares the voting result with all other CAVs:

$$V(x) = \arg \max_{p} \sum_{i=1}^{S} I(h_{j}(x) = p),$$
 (7)

where x is the operational data of all CAVs, $h_j(x)$ represents the current optimal passing order calculated by CAV_i based on the proposed MCTS method, $I(\cdot)$ denotes an indicator function, p is the possible candidate passing order calculated by all CAVs, and V(x) is the final passing order.

Note that each CAV executes the uniform passing order with the most votes. However, if two or more passing orders receive the same number of votes, we compare the objective values (3), and the passing order with the lower value is selected as the current optimal passing order.

3.4. Trajectory Control. After determining the passing order, the required acceleration adjustments and the desired arrival times to all conflict points are also determined. We must optimize the acceleration control of the CAVs to enable them to reach the conflict points at the desired times for passing through the unsignalized intersection safely and efficiently. For the longitudinal dynamics of CAVs in the same lane, we use a microscopic car-following model known as the intelligent driver model (IDM) [48]. The car-following model considers both the tendency to accelerate in free flow and decelerate to avoid colliding with the preceding vehicle. In the IDM model, the acceleration u_i of vehicle *i* is calculated by

$$u_i = u_{\max} \left[1 - \left(\frac{v_i}{v_0^{(i)}} \right)^{\delta} - \left(\frac{s * \left(v_i, \Delta v_i \right)}{s_i} \right)^2 \right], \tag{8}$$

where v_i is the speed of vehicle *i*, $v_0^{(i)}$ is the desired speed, s_i is the actual gap (distance to the preceding vehicle), Δv_i is the speed difference from the preceding vehicle, u_{max} is the maximum acceleration coefficient, δ is the acceleration exponent, and s * is the function for calculating the desired minimum gap, i.e.,

$$s * (\nu, \Delta \nu) = s_0^{(i)} + s_1^{(i)} \sqrt{\frac{\nu}{\nu_0^{(i)}}} + T^{(i)}\nu + \frac{\nu\Delta\nu}{2\sqrt{u_{\max}b_i}},$$
 (9)

where $s_0^{(i)}$ and $s_1^{(i)}$ are two distinct jam distances for vehicle *i*, $T^{(i)}$ is the safe time headway, and b_i is the desired deceleration coefficient.

When CAVs pass through the OPA into the EXA, the CAV required to give way performs the acceleration trajectory adjustments based on the solution of the optimization problem proposed in [49]; it performs the control in real time and keeps intervehicle safety gaps to arrive at the IPA on time.

4. Simulation Process

To verify the effectiveness of the proposed distributed MCTS-based cooperative driving strategy (D-MCTS), we consider the typical four-way, six-lane unsignalized intersection shown in Figure 1. We establish a SUMO (simulation of urban mobility) simulation environment and conduct simulation tests to compare our strategy with some existing classical intersection management strategies. The CR of the intersection is set to 200 m, which is within the effective communication of dedicated short-range communication technology (DSRC) [50]. The main parameters used in the simulation and the controller are given in Table 1 [51].

In the simulation, the traffic flow for CAVs is organized as follows: For the leftmost lane in each direction, 50% of the CAVs are programmed to turn left, while the remaining 50% proceed straight ahead. Similarly, half of the CAVs are designated to turn right in the rightmost lane, with the other half continuing straight. CAVs occupying the middle lane are exclusively allowed to travel straight. The arrival of approaching CAVs at the intersection is modeled as a Poisson process. These vehicles are assumed to enter each lane of the intersection entrances evenly, each at an initial speed of 10 m/s. To evaluate the efficacy of the proposed strategy under varying traffic conditions, the overall vehicle arrival rate is varied from 0 to 2 veh/s, equivalent to 0 to 600 veh/(lane * h).

When CAVs enter the control area, they share their operational information and use the no-conflict D-MCTS algorithms to coordinate their movements to pass through the intersection safely and in an orderly manner. The control algorithms are all executed in Python 3.8 and interact with SUMO through the Traci interface. In the present study, we reschedule the passing order of all CAVs within the control area at 2-second intervals.

TABLE 1: Parameters.

Parameters	Definition	Value
R_1	Radius of communication range	200 m
R_2	Radius of intersection physical area	30 m
$d_{\rm oba}$	Range of observation area	30 m
$d_{\rm opa}$	Range of optimization area	50 m
$d_{\rm exa}$	Range of execution area	90 m
w_l	Width of lane	3.5 m
l_{v}	Length of vehicle	5 m
$v_{\rm min}$	Minimum speed	0 m/s
$v_{\rm max}$	Maximum speed	12 m/s
$v_{\rm turn}$	Turning speed	6 m/s
$u_{\rm max}$	Maximum acceleration	$1.5 \mathrm{m/s^2}$
b	Desired deceleration	-3 m/s^2
$v_{ m ini}$	Initial arrival speed	10 m/s
ν_0	Desired speed	10 m/s
Т	Safe time headway	2 s
<i>s</i> ₀	Linear jam distance	5 m
λ	Arrival rate	0-2 veh/s
Δ_t	Simulation step size	0.1 s

To determine the optimal parameter settings for the D-MCTS cooperative driving strategy, this paper compares the total traffic delay D of the given n CAVs with the FCFS strategy. We define the decline rate of total traffic delay as η :

$$\eta = \frac{D_{\rm FCFS} - D_{D_MCTS}}{D_{\rm FCFS}},\tag{10}$$

where D_{FCFS} and $D_{D_{-\text{MCTS}}}$ are the total traffic delays of the FCFS based and D-MCTS based strategies, respectively.

First, to better understand the performance of the D-MCTS-based cooperative driving strategy under various traffic demands, we vary the vehicle arrival rate at the unsignalized intersection shown in Figure 1 to generate a variety of simulation scenarios and fix the computation time at 0.05 s. The weighing coefficients α_1 and α_2 of equation (3) are selected for sensitivity analysis under a moderate vehicle arrival rate of 400 veh/(lane * h). The experiments are conducted iteratively until the optimal driving strategy is identified, which maximizes the decline rate η , specifically at values of $\alpha_1 = 0.7$ and $\alpha_2 = 0.3$. Then, we vary ω and C from 0 to 1. Figure 7 shows the decline rates η of the D-MCTS-based strategy with 20 CAVs. This scenario is further investigated with different numbers of CAVs, and the findings are all consistent. The results show that despite the poor parameter settings, the D-MCTS strategy significantly improves results. The parameters ω and C are not particularly critical. However, they can still impact the balance of exploitation and exploration because we employ the heuristic rules in the simulation step to lessen the impact of random sampling. In certain cases, a larger value of C results in worse results because the agent has wasted too much processing time examining pointless nodes. However, some exploration is necessary because the decline rates with C = 0.25 are better than those with C = 0. Therefore, we set $\omega = 0.8$ and C = 0.25 in the rest of the experiments to maintain a good trade-off between exploitation and exploration.

To further determine the maximum computation time, we consider the relationship between the decline rate of total traffic delay and the number of searched nodes. For this



FIGURE 7: The decline rate η of the distributed MCTS-based strategy with different parameter settings for the intersection shown in Figure 1.

experiment, we choose the ideal parameter combination shown in Figure 7, and we change the arrival rates of the CAVs to generate a variety of driving scenarios with different numbers of CAVs, as well as recording the related decline rate and the number of searched nodes.

Figure 8 shows that the decline rate increases dramatically when the number of searched nodes increases from 1 to 400, after which it saturates. Therefore, the proposed D-MCTS strategy can give a sufficiently good passing order by searching 400 possible nodes for the considered scenarios. Generally, the decline rate increases gradually as the number of nodes searched by agents increases, and the more CAVs in the control area, the higher the decline rate for the same number of searched nodes. Note that the decline rates for scenarios with a small number of CAVs (30 CAVs) are low because the FCFS rule performs effectively in these straightforward scenarios. However, in situations with a larger number of CAVs (150 CAVs), there is not enough road space to adjust the vehicle passing order, and the decline rate is relatively small. For most intersection scenarios, 400 nodes can be searched within 0.05 s on our experimental device with an Intel i7 CPU and 16 GB RAM. For the following experiments, we set the maximum search time as 0.1 s to avoid errors caused by measurement and communication delays, and it is small enough for practical use.

To further delineate the difference between the FCFS strategy, the classical centralized MCTS-based strategy (C-MCTS), and our new proposed distributed MCTS-based strategy (D-MCTS), we study the established unsignalized intersection scenario with 20 vehicles. We calculate the objective values (3) of each strategy's optimal solution (passing order); see Table 2. The results indicate that the solution derived via the D-MCTS strategy closely aligns with the global-optimal solution obtained through the enumeration-based strategy. Notably, the computational time required for the two MCTS-based strategies is substantially lower. While the FCFS strategy exhibits the shortest computation time, its solution significantly diverges from the



FIGURE 8: The result of the decline rate η with respect to the number of searched nodes for the intersection shown in Figure 1.

optimal. Remarkably, the solution achieved by the D-MCTS strategy is ranked 190th among nearly 10 billion possible solutions, in stark contrast to the FCFS strategy's solution, which is ranked 3948842573rd.

5. Results and Analysis

In this section, we evaluate the performance of our newly proposed D-MCTS strategy compared to existing classical intersection management strategies under various vehicle arrival rates. These include the C-MCTS strategy, the FCFS strategy, the longest-queue-first (LQF) strategy, the actuated intersection control (AIC) strategy, and the traditional signal control strategy. All strategies' inflow traffic and time horizons are identical to ensure a fair comparison. Figure 9 illustrates the results, showcasing the average delay comparison across different arrival rates. To mitigate the effect of randomness in the outcomes and robustly compare the effectiveness of the strategies, simulations were conducted 50 times for each arrival rate scenario. In addition, Figure 9 includes the standard deviation of the average delay from the 50 simulation runs.

As depicted in Figure 9, there is a noticeable variation in intersection delays among the control strategies under the identical arrival-rate scenario. It is evident that as the number of CAVs at the intersection escalates, the D-MCTS strategy consistently exhibits the lowest delay among the six strategies, resulting in higher travel speed and throughput. Specifically, under a high traffic demand scenario with an arrival rate of 2 veh/s, the D-MCTS strategy (average delay: 17.1 s) outperforms the C-MCTS strategy by 1.7 s, the LQF strategy by 4.6 s, the AIC strategy by 5.6 s, the signal control strategy by 13.2 s, and the FCFS strategy by 28.2 s. This translates to improvements in traffic delay of 9%, 21.2%, 24.7%, 43.6%, and 62.3%, respectively. Moreover, when the arrival rate exceeds 1 veh/s, the FCFS strategy is most affected by changing traffic conditions due to its reliance on the arrival time of CAVs for priority assignment.

TABLE 2: Solution values of different cooperative driving strategies.

Strategies	Optimal solution (J)	Computation time	Ranking
Enumeration-based strategy	433.56	About 2 days	1
FCFS-based strategy	507.04	0.00001 s	3948842573
C-MCTS-based strategy	441.25	0.05 s	1051
D-MCTS-based strategy	435.03	0.05 s	190



FIGURE 9: Average delay of different driving strategies under different vehicle arrival rates.

Furthermore, the standard deviations indicate that the FCFS strategy is less effective than the two MCTS-based strategies in managing high arrival rates. Both MCTS strategies efficiently handle increased traffic density by balancing exploration and exploitation.

From a system optimization standpoint, the LQF strategy is often employed in intersection management, particularly within adaptive signal control systems. This strategy prioritizes longer queues at specific time points, which appears logical. However, this approach might not be the most effective in long-term scenarios. It also becomes apparent that the traditional signal control strategy does not excel in the comanagement of CAVs at intersections. A primary factor contributing to the heightened delays under the signal control strategy is its inefficient utilization of intersection space and time resources, a phenomenon observable even at relatively low traffic volumes. In addition, Figure 9 highlights a notable feature of the signal control strategy: its relatively small standard deviation. This indicates that signal control performance is largely unaffected by the randomness in traffic demand. Consequently, in certain situations, the signal control strategy may surpass the FCFS strategy in effectiveness.

Figure 10(a) illustrates the correlation between the average speed and arrival rate across the simulation area, employing six distinct control strategies. Notably, when the arrival rate is below 0.5 veh/s, the FCFS, LQF, C-MCTS, and D-MCTS strategies can maintain the average speed slightly above the initial speed. However, as the arrival rate increases, a decline in

average speed is observed. In this context, the D-MCTS strategy demonstrates a significant improvement compared to the other three strategies, with only a 35% reduction in average speed at an arrival rate of 2 veh/s. Furthermore, the AIC strategy exhibits superior performance over the traditional signal control strategy. This is attributed to the AIC strategy's ability to dynamically adjust traffic signal phases based on the density of vehicles at each entrance.

Figure 10(b) displays the average waiting times experienced under different control strategies at various arrival rates. Notably, CAVs managed by the D-MCTS strategy experience minimal waiting time when the arrival rate is below 1 veh/s. Furthermore, even as the arrival rate increases, the waiting time for CAVs under the D-MCTS strategy remains relatively low. In contrast, the FCFS strategy exhibits a substantial increase in waiting time as the arrival rate surpasses 0.5 veh/s. For the LQF strategy, it is observed that under high traffic demand, the LQF's passing pattern tends to resemble that of signal-controlled traffic, resulting in an average waiting time similar to that observed under the AIC strategy. Both the AIC and traditional signal control strategies demonstrate superior performance compared to the FCFS strategy when the arrival rate exceeds 1.25 veh/s.

Figure 10(c) clearly demonstrates that, in scenarios with the increased vehicular presence at the intersection, the D-MCTS strategy necessitates slighter acceleration adjustments compared to the FCFS and C-MCTS strategies. This characteristic significantly enhances the operational stability



FIGURE 10: Comparison results of significant traffic indicators of different driving strategies under different vehicle arrival rates. (a) Average speed. (b) Average waiting time. (c) Average number of adjustments. (d) Average CO_2 emission.

of CAVs under the D-MCTS strategy. Moreover, the reduced need for acceleration adjustments improves passenger comfort within the intersection. In conditions where the arrival rate reaches 2 veh/s, resulting in higher intersection congestion, the LQF strategy exhibits performance comparable to the D-MCTS strategy and markedly superior to the other two strategies mentioned above. In addition, it is observed that both signal-control strategies consistently require fewer acceleration adjustments, indicating that CAVs undergo less frequent acceleration and deceleration under signal-controlled modes compared to those managed by unsignalized intersection control strategies.

In Figure 10(d), we examine the environmental implications of CAVs navigating the intersection under various control strategies, focusing on the average CO_2 emissions. These emissions are quantified using the default emission function in the SUMO tool. It is observed that as traffic density intensifies, the LQF strategy increasingly demonstrates its effectiveness in reducing carbon emissions, closely followed by the D-MCTS strategy. Conversely, the CO_2 emissions escalate rapidly under the FCFS strategy and remain elevated under both signal control strategies. This trend suggests that frequent starting and stopping, regular acceleration and deceleration, and prolonged waiting times substantially increase carbon dioxide emissions, thereby exacerbating environmental pollution.

In addition, we analyze the impact of different control strategies on traffic throughput across varying traffic demand scenarios. A 100-minute traffic simulation was conducted for each arrival rate, with the comparative results detailed in Table 3. It is evident from these results that the proposed D-MCTS strategy significantly enhances traffic

Arrival rate (veh/s)	Strategy	Throughput (veh)
	FCFS	3220
	LQF	3170
0.5	C-MCTS	3200
0.5	D-MCTS	3230
	AIC	1670
	SIGNAL	1610
	FCFS	6040
	LQF	6070
1.0	C-MCTS	6130
1.0	D-MCTS	6160
	AIC	2980
	SIGNAL	2900
	FCFS	8340
	LQF	8510
1.5	C-MCTS	8670
1.5	D-MCTS	8860
	AIC	3890
	SIGNAL	3620
	FCFS	10860
	LQF	11210
2.0	C-MCTS	11440
2.0	D-MCTS	11800
	AIC	5870
	SIGNAL	5510

TABLE 3: Throughput under different arrival rates.







FIGURE 11: Average cross-section speed of the two MCTS-based cooperative driving strategies under different arrival rates. (a) Arrival rate = 0.5 veh/s. (b) Arrival rate = 1.0 veh/s. (c) Arrival rate = 1.5 veh/s. (d) Arrival rate = 2.0 veh/s.

throughput across all tested scenarios. As previously discussed, this improvement can be attributed to our distributed MCTS-based cooperative driving framework, which can explore and evaluate a broader range of CAV passing orders within the given planning time constraints compared to the centralized one.

As the number of CAVs at the unsignalized intersection increases, the performance of the FCFS strategy in terms of cooperation diminishes. However, the proposed D-MCTS strategy can always find a nearly global-optimal passing order regardless of the number of CAVs. Consequently, although the total delay inevitably escalates with an increasing number of CAVs, the D-MCTS strategy demonstrates a more pronounced capacity for improving the above key performance indicators.

To assess the effectiveness of the D-MCTS strategy in terms of driving stability, simulations were conducted under varying arrival rates. These simulations focused on monitoring the speed fluctuations of CAVs compared with the C-MCTS strategy at different cross sections of the intersection area. Induction loop detectors were strategically placed at distances of 50 m, 100 m, 150 m, and 200 m from the intersection center on each entrance lane of the east-west main road within the SUMO simulator. These detectors recorded the average speed and standard deviation of all CAVs passing through these cross sections. As presented in Figure 11, the findings indicate a superior performance of the D-MCTS strategy over the C-MCTS strategy across various traffic demand scenarios. Notably, the D-MCTS strategy ensures smoother operation speeds at different cross sections, leading to reduced speed volatility, enhanced operational stability, and quicker passage through the unsignalized intersection. This results in an overall improvement in passenger comfort.

6. Conclusions

This study proposes a distributed MCTS-based cooperative driving strategy for CAVs at unsignalized intersections called D-MCTS. It integrates root parallelization MCTS with the majority voting rule to implement the distributed cooperation, aiming to explore and evaluate the feasible vehicle passing orders as many as possible within the limited planning time to find a nearly global-optimal passing order enabling CAVs to minimize traffic delays while making the slightest acceleration adjustments. In addition, the research develops a task-area partition framework to decompose the mission of cooperative driving into three main tasks: vehicle information sharing, passing order optimization, and trajectory control.

In a comparative analysis conducted within the SUMO simulation environment, the efficacy of the proposed D-MCTS strategy was validated against five other driving strategies. During heavy traffic flow, the D-MCTS strategy reduced the average delay for CAVs by 9.0% to 62.3%. Furthermore, there was only about a 35% decrease in average speed, and the average waiting time remained minimal. Notably, the average number of acceleration adjustments was lower than that of the FCFS, LQF, and C-MCTS strategies, indicating a significant improvement in CAV operational stability. Regarding carbon emissions, the

D-MCTS strategy was outperformed only by the LQF strategy. In addition, the D-MCTS strategy effectively enhanced traffic throughput across all scenarios. The simulation results demonstrate that the new D-MCTS strategy noticeably improved efficiency, safety, comfort, and emissions.

However, the present study only considered a single scenario of a two-way, three-lane unsignalized intersection in a pure CAV environment, for which the traffic environment was limited. Future work should consider more complex traffic scenarios, such as multiple unsignalized intersection networks at different CAV penetration rates, which would lead to more findings on how the proposed distributed cooperative driving strategy affects mixed traffic streams.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

The authors thank the Transportation Research Center in Shanghai Jiao Tong University for their technical support and ChatGPT for language polishing assistance.

References

- D. J. Fagnant and K. Kockelman, "Preparing a nation for autonomous vehicles: opportunities, barriers and policy recommendations," *Transportation Research Part A: Policy* and Practice, vol. 77, pp. 167–181, 2015.
- [2] D. G. Dorrell, A. Vinel, and D. P. Cao, "Connected vehiclesadvancements in vehicular technologies and informatics," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 12, pp. 7824–7826, 2015.
- [3] W. Do, O. M. Rouhani, and L. Miranda-Moreno, "Simulation-based connected and automated vehicle models on highway sections: a literature review," *Journal of Advanced Transportation*, vol. 2019, Article ID 9343705, 14 pages, 2019.
- [4] S. Aoki and R. Rajkumar, "Safe intersection management with cooperative perception for mixed traffic of human-driven and autonomous vehicles," *Ieee Open Journal of Vehicular Technology*, vol. 3, pp. 251–265, 2022.
- [5] K. Sjöberg, P. Andres, T. Buburuzan, and A. Brakemeier, "Cooperative intelligent transport systems in europe: current deployment status and outlook," *IEEE Vehicular Technology Magazine*, vol. 12, no. 2, pp. 89–97, 2017.
- [6] E. Moradi-Pari, H. N. Mahjoub, H. Kazemi, Y. P. Fallah, and A. Tahmasbi-Sarvestani, "Utilizing model-based communication and control for cooperative automated vehicle applications," *Ieee Transactions on Intelligent Vehicles*, vol. 2, no. 1, pp. 38–51, 2017.
- [7] C. Y. Liu, Z. H. Sheng, S. K. Chen, H. T. Shi, and B. Ran, "Longitudinal control of connected and automated vehicles among signalized intersections in mixed traffic flow with deep

- [8] T. Takahama and D. Akasaka, "Model predictive control approach to design practical adaptive cruise control for traffic jam," *International Journal of Automotive Engineering*, vol. 9, no. 3, pp. 99–104, 2018.
- [9] Y. Guo and J. Q. Ma, "Leveraging existing high-occupancy vehicle lanes for mixed-autonomy traffic management with emerging connected automated vehicle applications," *Transportmetrica a-Transport Science*, vol. 16, no. 3, pp. 1375–1399, 2020.
- [10] S. Knorn, A. Donaire, J. C. Agüero, and R. H. Middleton, "Passivity-based control for multi-vehicle systems subject to string constraints," *Automatica*, vol. 50, no. 12, pp. 3224– 3230, 2014.
- [11] P. Hang, C. Lv, C. Huang, J. C. Cai, Z. X. Hu, and Y. Xing, "An integrated framework of decision making and motion planning for autonomous vehicles considering social behaviors," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 12, pp. 14458–14469, 2020.
- [12] Y. Zheng, Y. G. Bian, S. B. E. Li, and S. Li, "Cooperative control of heterogeneous connected vehicles with directed acyclic interactions," *Ieee Intelligent Transportation Systems Magazine*, vol. 13, no. 2, pp. 127–141, 2021.
- [13] Y. F. Yao, Y. Zhou, J. X. Xia, X. Wang, X. T. Li, and B. Ran, "Rolling-horizon-based strategy of fully cooperative traffic under signalized intersections," *Computer-Aided Civil and Infrastructure Engineering*, vol. 38, no. 4, pp. 454–469, 2023.
- [14] P. Ha, S. K. Chen, J. Q. Dong, and S. Labi, "Leveraging vehicle connectivity and autonomy for highway bottleneck congestion mitigation using reinforcement learning," *Transportmetrica a-Transport Science*, vol. 0, no. 1, pp. 1–26, 2023.
- [15] J. Li, X. D. Wu, M. Xu, and Y. G. Liu, "Deep reinforcement learning and reward shaping based eco-driving control for automated HEVs among signalized intersections," *Energy*, vol. 251, Article ID 123924, 2022.
- [16] Y. Li, Y. Ma, and Z. Chen, "How can connected and automated vehicles improve merging efficiency at freeway onramps?" *Transportmetrica: Transportation Science*, vol. 20, no. 2, pp. 1–27, 2022.
- [17] B. Leng, Y. Z. Yu, M. Liu, L. Cao, X. Yang, and L. Xiong, "Deep reinforcement learning-based drift parking control of automated vehicles," *Science China Technological Sciences*, vol. 66, no. 4, pp. 1152–1165, 2023.
- [18] J. Goyani, N. Gore, and S. Arkatkar, "Modeling crossing conflicts at unsignalized T-intersections under heterogeneous traffic conditions," *Journal of Advanced Transportation*, vol. 2022, Article ID 9965733, 12 pages, 2022.
- [19] M. Khayatian, M. Mehrabian, E. Andert et al., "A survey on intersection management of connected autonomous vehicles," *Acm Transactions on Cyber-Physical Systems*, vol. 4, no. 4, pp. 1–27, 2020.
- [20] Z. J. Wang, X. M. Chen, P. Wang, M. X. Li, Y. J. X. Ou, and H. Zhang, "A decision-making model for autonomous vehicles at urban intersections based on conflict resolution," *Journal of Advanced Transportation*, vol. 2021, Article ID 8894563, 12 pages, 2021.
- [21] E. F. Ozioko, J. Kunkel, and F. Stahl, "Road intersection coordination scheme for mixed traffic (Human-Driven and driverless vehicles): a systematic review," *Journal of Advanced Transportation*, vol. 2022, Article ID 2951999, 15 pages, 2022.
- [22] G. N. Bifulco, A. Coppola, A. Petrillo, and S. Santini, "Decentralized cooperative crossing at unsignalized intersections via

vehicle-to-vehicle communication in mixed traffic flows," Journal of Intelligent Transportation Systems, vol. 28, no. 2, pp. 1–26, 2022.

- [23] P. Hang, C. Huang, Z. X. Hu, and C. Lv, "Driving conflict resolution of autonomous vehicles at unsignalized intersections: a differential game approach," *Ieee-Asme Transactions on Mechatronics*, vol. 27, no. 6, pp. 5136–5146, 2022.
- [24] Y. Meng, L. Li, F. Y. Wang, K. Q. Li, and Z. H. Li, "Analysis of cooperative driving strategies for nonsignalized intersections," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 4, pp. 2900–2911, 2018.
- [25] L. Li and F. Y. Wang, "Cooperative driving at blind crossings using intervehicle communication," *IEEE Transactions on Vehicular Technology*, vol. 55, no. 6, pp. 1712–1724, 2006.
- [26] H. L. Xu, Y. Zhang, C. G. Cassandras, L. Li, and S. Feng, "A bilevel cooperative driving strategy allowing lane changes," *Transportation Research Part C: Emerging Technologies*, vol. 120, Article ID 102773, 2020.
- [27] H. L. Xu, S. Feng, Y. Zhang, and L. Li, "A grouping-based cooperative driving strategy for CAVs merging problems," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 6, pp. 6125–6136, 2019.
- [28] K. Dresner and P. Stone, *Multiagent Traffic Management*, ACM, New York, NY, USA.
- [29] K. Dresner and P. Stone, "A multiagent approach to autonomous intersection management," *Journal of Artificial Intelligence Research*, vol. 31, pp. 591–656, 2008.
- [30] M. Choi, A. Rubenecia, and H. H. Choi, "Reservation-based cooperative traffic management at an intersection of multilane roads," in *Proceedings of the 2018 32nd International Conference on Information Networking (Icoin)*, pp. 456–460, Chiang Mai, Thailand, January 2018.
- [31] A. A. Malikopoulos, C. G. Cassandras, and Y. J. Zhang, "A decentralized energy-optimal control framework for connected automated vehicles at signal-free intersections," *Automatica*, vol. 93, pp. 244–256, 2018.
- [32] Y. Zhang and C. G. Cassandras, "Decentralized optimal control of Connected Automated Vehicles at signal-free intersections including comfort-constrained turns and safety guarantees," *Automatica*, vol. 109, Article ID 108563, 2019.
- [33] P. F. Li and X. S. Zhou, "Recasting and optimizing intersection automation as a connected-and-automated-vehicle (CAV) scheduling problem: a sequential branch-and-bound search approach in phase-time-traffic hypernetwork," *Transportation Research Part B: Methodological*, vol. 105, pp. 479– 506, 2017.
- [34] S. A. Fayazi and A. Vahidi, "Mixed-integer linear programming for optimal scheduling of autonomous vehicle intersection crossing," *Ieee Transactions on Intelligent Vehicles*, vol. 3, no. 3, pp. 287–299, 2018.
- [35] H. L. Xu, Y. Zhang, L. Li, and W. X. Li, "Cooperative driving at unsignalized intersections using tree search," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 11, pp. 4563–4571, 2020.
- [36] Y. Zhang and C. G. Cassandras, "A decentralized optimal control framework for connected automated vehicles at urban intersections with dynamic resequencing," in *Proceedings of the 2018 Ieee Conference on Decision and Control (Cdc)*, pp. 217–222, Miami, FL, USA, December 2018.
- [37] Z. Y. Deng, Y. J. Shi, Q. M. Han, L. L. Lv, and W. M. Shen, "A conflict duration graph-based coordination method for connected and automated vehicles at signal-free intersections," *Applied Sciences*, vol. 10, no. 18, p. 6223, 2020.

- [38] C. Y. Chen, Q. Xu, M. C. Cai, J. W. Wang, J. Q. Wang, and K. Q. Li, "Conflict-free cooperation method for connected and automated vehicles at unsignalized intersections: graphbased modeling and optimality analysis," *IEEE Transactions* on Intelligent Transportation Systems, vol. 23, no. 11, pp. 21897–21914, 2022.
- [39] G. Lu, Z. Shen, X. Liu, Y. Nie, and Z. Xiong, "Are autonomous vehicles better off without signals at intersections? A comparative computational study," *Transportation Research Part B: Methodological*, vol. 155, pp. 26–46, 2022.
- [40] D. R. Morrison, S. H. Jacobson, J. J. Sauppe, and E. C. Sewell, "Branch-and-bound algorithms: a survey of recent advances in searching, branching, and pruning," *Discrete Optimization*, vol. 19, pp. 79–102, 2016.
- [41] C. Wang, L. J. Lu, J. Lu, and T. Wang, "Correlation between crash avoidance maneuvers and injury severity sustained by motorcyclists in single-vehicle crashes," *Traffic Injury Prevention*, vol. 17, no. 2, pp. 188–194, 2016.
- [42] M. Swiechowski, K. Godlewski, B. Sawicki, and J. Mandziuk, "Monte Carlo Tree Search: a review of recent modifications and applications," *Artificial Intelligence Review*, vol. 56, no. 3, pp. 2497–2562, 2023.
- [43] C. B. Browne, E. Powley, D. Whitehouse et al., "A survey of Monte Carlo tree search methods," *Ieee Transactions on Computational Intelligence and Ai in Games*, vol. 4, no. 1, pp. 1–43, 2012.
- [44] G. M. J. B. Chaslot, M. H. M. Winands, and H. J. van den Herik, "Parallel monte-carlo tree search," *Computers and Games*, vol. 5131, p. 60, 2008.
- [45] E. Steinmetz and M. Gini, "More trees or larger trees: parallelizing Monte Carlo tree search," *Ieee Transactions on Games*, vol. 13, no. 3, pp. 315–320, 2021.
- [46] K. Kurzer, C. Hörtnagl, and J. M. Zöllner, "Parallelization of monte carlo tree search in continuous domains," 2020, https://arxiv.org/abs/2003.13741.
- [47] Y. Soejima, A. Kishimoto, and O. Watanabe, "Evaluating root parallelization in go," *Ieee Transactions on Computational Intelligence and Ai in Games*, vol. 2, no. 4, pp. 278–287, 2010.
- [48] L. Zhu, L. J. Lu, X. N. Wang, C. M. Jiang, and N. F. Ye, "Operational characteristics of mixed-autonomy traffic flow on the freeway with on- and off-ramps and weaving sections: an RL-based approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 8, pp. 13512–13525, 2022.
- [49] H. L. Xu, C. G. Cassandras, L. Li, and Y. Zhang, "Comparison of cooperative driving strategies for CAVs at signal-free intersections," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 7614–7627, 2022.
- [50] J. B. Kenney, "Dedicated short-range communications (DSRC) standards in the United States," *Proceedings of the IEEE*, vol. 99, no. 7, pp. 1162–1182, 2011.
- [51] K. M. Williams, V. G. Stover, K. K. Dixon, and P. Demosthenes, Access Management Manual, 2014, https://trid.trb.org/view/ 1341995.