# Comparison of Cooperative Driving Strategies for CAVs at Signal-Free Intersections

Huile Xu, Christos G. Cassandras, *Life Fellow, IEEE*, Li Li, *Fellow, IEEE*, and Yi Zhang, *Member, IEEE*

***Abstract*— The properties of cooperative driving strategies for planning and controlling Connected and Automated Vehicles (CAVs) at intersections range from some that achieve highly efficient coordination performance to others whose implementation is computationally fast. This paper comprehensively compares the performance of four representative strategies in terms of travel time, energy consumption, computation time, and fairness under different conditions, including the geometric configuration of intersections, asymmetry in traffic arrival rates, and the relative magnitude of these rates. Our simulation-based study has led to the following conclusions: 1) The Monte Carlo Tree Search (MCTS)-based strategy achieves the best traffic efficiency and has great performance in fuel consumption; 2) MCTS and Dynamic Resequencing (DR) strategies both perform well in all metrics of interest. If the computation budget is adequate, the MCTS strategy is recommended; otherwise, the DR strategy is preferable; 3) An asymmetric intersection has a noticeable impact on the strategies, whereas the influence of the arrival rates can be neglected. When the geometric shape is asymmetrical, the modified First-In-First-Out (FIFO) strategy significantly outperforms the FIFO strategy and works well when the traffic demand is moderate, but their performances are similar in other situations; and 4) Improving traffic efficiency sometimes comes at the cost of fairness, but the DR and MCTS strategies can be adjusted to realize a better trade-off between various performance metrics by appropriately designing their objective functions.**

Manuscript received June 16, 2020; revised December 21, 2020 and February 23, 2021; accepted April 2, 2021. The work of Huile Xu was supported in part by the National Key Research and Development Program of China under Grant 2018YFB1600600; in part by the NSF under Grant ECCS-1931600, Grant DMS-1664644, and Grant CNS-1645681; in part by the Air Force Office of Scientific Research (AFOSR) under Grant FA9550-19-1-0158; in part by the Advanced Research Projects Agency–Energy's (ARPA-E's) NEXTCAR Program under Grant DE-AR0000796; and in part by the MathWorks. The work of Christos G. Cassandras was supported in part by the NSF under Grant ECCS-1931600, Grant DMS-1664644, and Grant CNS-1645681; in part by the AFOSR under Grant FA9550-19-1-0158; in part by the ARPA-E's NEXTCAR Program under Grant DE-AR0000796; and in part by the MathWorks. The work of Li Li and Yi Zhang was supported by the National Key Research and Development Program of China under Grant 2018YFB1600600. The Associate Editor for this article was L. Li. *(Corresponding Author: Yi Zhang.)*

Huile Xu is with the BNRist, Department of Automation, Tsinghua University, Beijing 100084, China, also with the Division of Systems Engineering, Boston University, Brookline, MA 02446 USA, and also with the Center for Information and Systems Engineering, Boston University, Brookline, MA 02446 USA (e-mail: hl-xu16@mails.tsinghua.edu.cn).

Christos G. Cassandras is with the Division of Systems Engineering, Boston University, Brookline, MA 02446 USA, and also with the Center for Information and Systems Engineering, Boston University, Brookline, MA 02446 USA (e-mail: cgc@bu.edu).

Li Li is with the BNRist, Department of Automation, Tsinghua University, Beijing 100084, China (e-mail: li-li@tsinghua.edu.cn).

Yi Zhang is with the BNRist, Department of Automation, Tsinghua University, Beijing 100084, China, and also with the Tsinghua-Berkeley Shenzhen Institute (TBSI), Tsinghua University, Shenzhen 518055, China (e-mail: zhyi@tsinghua.edu.cn).

Digital Object Identifier 10.1109/TITS.2021.3071456

***Index Terms*— Connected and automated vehicles (CAVs), cooperative driving strategy, crossing sequence.**

## I. Introduction

INTERSECTIONS are the main bottlenecks for urban traffic. As reported in [1], congestion in these areas causes substantial economic loss to society and significantly increases the travel time of drivers. Coordination and control problems at intersections are challenging in terms of safety, traffic efficiency, and energy consumption [1], [2].

The emergence of Connected and Automated Vehicles (CAVs) is believed to be a promising way of improving safety, traffic efficiency as well as reducing energy consumption. With the aid of vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication, CAVs can obtain real-time operational data from neighboring CAVs and communicate with the infrastructure [3]. These technologies have made it possible to plan better trajectories for CAVs through optimal control methods, as well as implement these trajectories in real time.

In recent years, various cooperative driving strategies have been proposed to achieve optimal coordination for CAVs driving through signal-free intersections. The goal of these strategies is to minimize one or several objectives by scheduling both the crossing order and the control inputs (speed, acceleration) of all CAVs. Thus, cooperative driving strategies mainly consist of two parts: 1) a scheduling problem in terms of crossing sequences and controllable arrival times at conflict areas; and 2) an optimal control problem in terms of control inputs. This paper focuses on the first problem and divides the existing strategies into two kinds from the perspective of crossing sequences, i.e., cooperative driving strategies without resequencing and cooperative driving strategies with resequencing.

Cooperative driving strategies *without* resequencing mainly refer to the First-In-First-Out (FIFO) approach where we directly determine the crossing sequence according to the order of CAVs entering a control zone (defined as an area around the intersection within which V2I communication is possible). Any new arriving vehicle does not influence the crossing sequence already determined for previous CAVs. For example, Stone *et al.* proposed an autonomous intersection management cooperative driving strategy that divides the intersection into grids (resources) and assigns these grids to CAVs in a FIFO manner [4], [5]. Malikopoulos *et al.* designed a decentralized time-then-energy optimal control framework for CAVs at intersections where they obtained the desired optimal arrival times of CAVs based on a FIFO crossing sequence and derived the energy-optimal analytical solution for controlling CAVs to arrive at the intersection's conflict (merging) zone at these prescribed arrival times [6]. Zhang and Cassandras further extended the work by including all possible turns and considering the joint energy-time-optimal solution [7]. In addition, they incorporated safe distance constraints within the control zone and passenger comfort within the conflict zone into the trajectory optimization framework. However, recent studies have shown that the performance of the FIFO crossing sequence may be far from the optimal solution in at least some cases [8].

Cooperative driving strategies *with* resequencing aim to find a better crossing sequence for CAVs within the control zone. One of the prevailing ideas is to formulate an optimization problem whose decision variables are crossing sequences and control inputs. Specifically, binary variables are introduced to represent the crossing priority between any two CAVs, which leads to Mixed-Integer Linear Programming (MILP) problems [9], [10]. However, MILP problems are NP-hard, i.e., their computation time increases exponentially with the number of considered CAVs. Alternatively, it has been shown that the problem may be treated as a tree search problem where each tree node represents a special crossing sequence [11]. The equivalent objective is to find a leaf node that corresponds to the optimal solution. However, this approach faces similar computational disadvantages as MILP-based strategies. Although techniques such as grouping methods [12] or pruning [13] have been proposed to reduce the size of the original problem or to accelerate the search process, it is still hard to obtain a real-time solution for complicated driving scenarios that arise, for example, in multi-lane intersections. To overcome the above shortcomings, there have been several recent studies on this topic. For example, Xu *et al.* proposed a Monte Carlo Tree Search (MCTS)-based strategy where they used the MCTS to guide the search process and determine as many promising crossing sequences as possible within a limited computation budget [8]. By performing comparisons with the results of exhaustive searching (whenever possible), they demonstrated that the MCTS strategy can always find a near-optimal solution, even for complicated multi-lane intersections where the search space is enormous [8]. Following a very different approach, Zhang and Cassandras designed a Dynamic Resequencing (DR) scheme to optimize the crossing sequence [14]. Rather than periodically replanning crossing sequences for all CAVs as in the MCTS approach, the DR strategy keeps the original crossing sequence unchanged and updates it only when a new CAV arrives by inserting it into a suitable position within the original sequence so as maximally improve performance. Nevertheless, due to the different models and simulation settings used by different researchers, we still lack a comprehensive comparative performance evaluation for these strategies under different driving scenarios.

To analyze the relative advantages and disadvantages of different cooperative driving strategies, we have selected four representative types of such strategies: the MCTS strategy [8], the DR strategy [14], the commonly used FIFO strategy, and a modified FIFO-based strategy. This paper first applies these strategies to a typical signal-free single-lane intersection with the same arrival rate at each lane and a symmetric geometrical shape. Then, we vary the length of different lanes and associated arrival rates in order to investigate the impact of asymmetrical intersection geometries and asymmetrical arrival rates on these strategies, respectively. In addition to performance metrics such as travel time and energy, we also compare the computation time of different strategies and the number of crossing sequences they have considered during their computation time. Finally, we discuss the drawback often caused by resequencing, i.e., unfairness across the different traffic arrival lanes, and introduce a balancing factor to the DR strategy so as effectively control the trade-off between fairness and efficiency.

The main contributions of this paper are: 1) to comprehensively evaluate and compare the performance of representative state-of-the-art cooperative driving strategies; 2) to analyze the influence of asymmetrical arrival rates and intersection geometries on these strategies; and 3) to explore the trade-off between different performance metrics and to propose improvements to existing strategies based on resequencing.

The paper is organized as follows. *Section II* formulates the optimal control problem of controlling CAVs passing through a signal-free intersection safely. *Section III* briefly reviews the four cooperative driving strategies to be compared. Then, in *Section IV* we conduct a series of experiments to compare their performance under different simulation settings. *Section V* discusses the trade-off between fairness and efficiency. Finally, *Section VI* gives concluding remarks.

## II. Problem Formulation

Figure 1 shows a typical intersection configuration with a single lane in each direction. The area within the circle is called the *Control Zone*, while the shadowed area is called the *Conflict Zone* where lateral collisions may happen. The road segment from the entry of the Control Zone to the entry of the Conflict Zone is referred to as a control zone segment, and its length is denoted by $L_i$, $i \in \{1, 2, 3, 4\}$. The value of $L_i$ is usually associated with the communication range of road-side infrastructure equipment (often referred to as a road-side unit). If all $L_i$ are equal, then the intersection is symmetrical; otherwise, it is an asymmetrical intersection. To improve space utilization, we divide the Conflict Zone into several subzones. For example, the Conflict Zone in Fig. 1 is divided into 4 Conflict Subzones, which are labeled Conflict Subzone 1 through Conflict Subzone 4. After this division, CAVs that pass through different subzones can cross the intersection at the same time.

When a CAV enters the Control Zone, we assign it a unique identity $i$, labeling it as the $i$th CAV. Then, we use the sequence $Z_i$ to denote the Conflict Subzones that CAV $i$ will pass through. For example, $Z_i = \{4, 1\}$ in Fig. 1 means CAV $i$ will pass through Conflict Subzone 4 and then Conflict Subzone 1 in sequence.

To simplify the problem, we adopt the following assumptions:

- Each vehicle instantly shares its complete driving state (position, velocity, etc.) and intentions with other CAVs via V2V communication (or V2I then I2V).

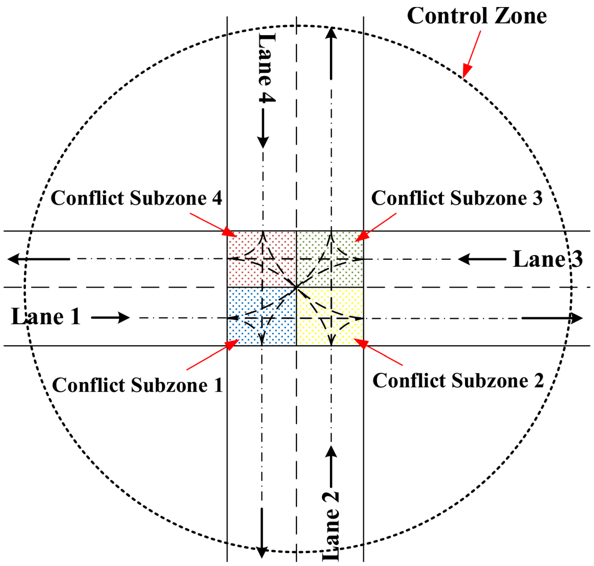



Fig. 1. A typical intersection.

- Similar to [6] and [15], the velocities of CAVs are constant when passing through the Conflict Zone.

As already mentioned, cooperative driving strategies consist of two parts. We obtain the optimal crossing sequence and corresponding arrival time by solving the scheduling problem in the first part. Then, the arrival time is utilized as the terminal time for solving an optimal control problem in the second part, through which a CAV's inputs are determined. It is worth noting that some studies combine these two parts into a single optimization problem [16], but the computation time for solving such a problem is prohibitively large even when tools such as model predictive control are employed [17]. Moreover, it is hard to extend this method to problems with complicated objective functions and vehicle dynamics. On the other hand, if the crossing sequence is given in advance, then it is possible to efficiently design a decentralized optimal control problem for jointly optimizing the arrival times and control inputs of each CAV as in [7] and [18]. We regard these methods as extensions of the basic cooperative coordination problem and will, therefore, not consider them here.

### A. Scheduling Problem in Terms of Arrival Times and Crossing Sequences

Let $a_{i,z}$ denote the desired arrival time to the Conflict Subzone $z$ for CAV $i$, and $\sigma_{i,z}$ is the minimum arrival time to the Conflict Subzone $z$ when CAV $i$ travels with the maximum velocity and maximum acceleration. It is clear that $\sigma_{i,z}$ is the fixed lower bound for $a_{i,z}$. Let $Z_i^1$ be the first element in $Z_i$, e.g., $Z_i^1 = 4$ when $Z_i = \{4, 1\}$. Thus, $a_{i,Z_i^1}$ is the arrival time at the first Conflict Subzone that CAV $i$ will pass through. We also use $C_{i,z}$ to include the set of indices of all CAVs that may collide with CAV $i$ in the Conflict Subzone $z$. Once $i$ is known, $Z_i^1$ and $C_{i,z}$ are fully determined.

We introduce binary variables $\boldsymbol{b} = [b_{1,2}, b_{1,3}, \ldots, b_{n-1,n}]$ to represent crossing sequences, where $n$ is the number of CAVs currently in the Control Zone and not yet at the Conflict Zone. We use $b_{i,j} = 1$ to indicate that CAV $i$ is assigned to cross the Conflict Zone before CAV $j$ for every $j \in C_{i,z}$, $z \in Z_i$, such that $j$ may conflict with $i$; otherwise, $b_{i,j} = 0$ indicating that CAV $j$ has higher crossing priority. Therefore, the vector $\boldsymbol{b}$ always contains elements which allow us to interpret it as a crossing sequence in the form of a string. For example, $\boldsymbol{b} = [b_{1,2}, b_{1,3}, b_{2,3}] = [1, 1, 0]$ implies the crossing sequence is 132. It is also clear that $b_{i,j} = 1 - b_{j,i}$, so we omit all $b_{j,i}$ with $j \geq i$ in the definition of $\boldsymbol{b}$.

In addition, let $\boldsymbol{a}$ denote the vector of all $a_{i,z}$. where $a_{i,z}$ is a continuous variable whose range is between $\sigma_{i,z}$ and $\infty$. We can then formulate the following optimization problem:

$$\min_{\boldsymbol{a},\boldsymbol{b}} \sum_{i=1}^{n} (a_{i,Z_i^1} - \sigma_{i,Z_i^1}) \tag{1a}$$

$$\text{s.t. } a_{i,z} \geq \sigma_{i,z}, \quad i = 1, \ldots, n, \ z \in Z_i \tag{1b}$$

$$a_{i,Z_i^1} - a_{p,Z_i^1} \geq \Delta t_p, \quad i = 1, \ldots, n \tag{1c}$$

$$a_{i,z} - a_{j,z} + M \cdot b_{i,j} \geq \Delta t_j, \quad i \in N, \ z \in Z_i, \ j \in C_{i,z} \tag{1d}$$

$$a_{j,z} - a_{i,z} + M \cdot (1 - b_{i,j}) \geq \Delta t_i, \quad i \in N, \quad z \in Z_i, \quad j \in C_{i,z} \tag{1e}$$

$$N = \{1, 2, \ldots, n\}, \tag{1f}$$

$$b_{i,j} \in \{0, 1\}. \tag{1g}$$

Constraints (1c) capture the safety rear-end constraints for all CAVs in the same lane, and CAV $p$ is the CAV physically preceding (ahead of) CAV $i$. $\Delta t_p$ is the safety time headway between two CAVs and is related to the type and movement of CAV $p$. Constraints (1d) and (1e) are safety lateral constraints for CAVs $i$ and $j$ to ensure that there is no more than one vehicle in any Conflict Subzone at any time. $M$ is a sufficiently large number such that if $b_{i,j} = 1$, then inequality (1d) must hold (due to the large value of $M$). It follows that inequality (1e) takes on the same form as (1c). Thus, if $b_{i,j} = 1$, CAV $i$ is prioritized to cross the Conflict Zone earlier than CAV $j$.

*Remark: Regarding the selection of a value for $M$, it is straightforward to derive a finite lower bound for it such that any value greater than this lower bound may be chosen. In particular, let $a_{\max,z}$ be the arrival time at Conflict Subzone $z$ when a CAV starts at a Control Zone entry point with the minimum initial velocity. For simplicity, define $\Delta t = \max(\Delta t_i), \forall i$. Then, for any $a_{i,z}$ we have $a_{i,z} = \max(a_{k,z} + \Delta t_k, \sigma_{i,z})$ where $k$ is the last CAV passing through Conflict Subzone $z$ prior to CAV $i$. Since, $\sigma_{i,z} \leq a_{\max,z}, \forall i$, we have $a_{i,z} \leq a_{\max,z} + (n-1)\Delta t$ when CAV $i$ is the nth CAV in the crossing sequence. Then, since $a_{j,z} > 0$, $\forall j$, we have*

$$a_{i,z} - a_{j,z} + \Delta t < a_{\max,z} + (n-1)\Delta t + \Delta t < a_{\max,z} + N\Delta t$$

*where $N$ is the capacity of the intersection in terms of the number of CAVs it can accommodate.*

By solving the optimization problem (1) to obtain a solution $(\boldsymbol{a}, \boldsymbol{b})$, we get the optimal crossing sequence (given by $\boldsymbol{b}^*$) and the desired arrival times for all CAVs. Clearly, (1) is a MILP problem that is NP-hard. The commonly used methods for solving MILP problems are branch and bound algorithms with a worst-case running time of $O(b^d)$, where $b$ is the branching factor and $d$ is the search depth (relevant to the number of CAVs). Thus the computation time for solving this kind of problem increases exponentially with the number of CAVs.

To indirectly solve the problem, [6] and [12] pointed out that we can determine the crossing sequence first and then the primal problem reduces to a linear programming problem that can be easily solved. To be more specific, after we specify the value of $\boldsymbol{b}$, then (1d) and (1e) become linear constraints without binary variables and the problem (1) only has one decision variable $\boldsymbol{a}$. Moreover, we can design a simple iterative algorithm (**Algorithm 1**) to solve the problem (1) given $\boldsymbol{b}$ and derive the desired arrival times for all CAVs with a time complexity $O(n)$.

**Algorithm 1** Crossing Sequence to Trajectory Interpretation

**Input:** A possible value of $\boldsymbol{b}$

**Output:** The total delay $J$ of the covered vehicles and their arrival times $a_{i,z}$

```
Interpret b as a crossing sequence in the form of a string S.
for each i ∈ [1, length(S)] do
  Let j = S(i), and S(i) is the ith element of S.
  for each z ∈ Z_j do
    CAV k is the last CAV passing through subzone z prior to CAV j
    if CAV k exists then
      a_{j,z} = max(σ_{j,z}, a_{k,z} + Δt_k)
    else
      a_{j,z} = σ_{j,z}
    end if
  end for
  Adjust a_{j,z} according to the constraint: the velocity of CAV j in the Conflict Zone is constant.
end for
J = Σ_j (a_{j,Z_j^1} − σ_{j,Z_j^1})
```

Based on this idea, the original problem is transformed into a problem of finding the optimal crossing sequence for improving traffic efficiency. In recent years, there have been many state-of-the-art studies on this topic, which will be introduced in detail in the next section.

### *B. Optimal Control Problem in Terms of Control Inputs*

After determining the desired arrival times, we need to plan control inputs for optimally controlling CAVs so that they arrive at the Conflict Zone at the desired time and at the same time minimize a specific objective. Aside from traffic efficiency, energy consumption is a performance metric of interest. Since the energy consumption rate of CAV $i$ is a function of its control inputs and monotonically increasing with the acceleration $u_i$, we formulate the following optimal control problem solved by each CAV $i$ in a decentralized fashion:

$$\min_{u_i(t)} \int_{t_i^0}^{a_{i,Z_i^1}} \mathcal{C}(u_i(t))dt \tag{2a}$$

$$\text{s.t. } \dot{x}_i(t) = v(t),\ \dot{v}_i(t) = u(t), \quad t \in [t_i^0, a_{i,Z_i^1}] \tag{2b}$$

$$x_i(t_i^0) = 0, \quad v_i(t_i^0) = v_i^0, \tag{2c}$$

$$x_i(a_{i,Z_i^1}) = L, \tag{2d}$$

$$v_i(a_{i,Z_i^1}) = v_i^f, \tag{2e}$$

$$x_p(t) - x_i(t) \geq l + v_i(t)h_s, \quad t \in [t_i^0, a_{i,Z_i^1}] \tag{2f}$$

$$v_{\min,i} \leq v_i(t) \leq v_{\max,i}, \quad t \in [t_i^0, a_{i,Z_i^1}] \tag{2g}$$

$$a_{\min,i} \leq u_i(t) \leq a_{\max,i}, \quad t \in [t_i^0, a_{i,Z_i^1}] \tag{2h}$$

where $\mathcal{C}(\cdot)$ is a strictly increasing function of its argument, e.g., $\mathcal{C}(u_i(t)) = \frac{1}{2}u_i^2(t)$. Constraints (2b) consist of the vehicle dynamics where $x_i(t)$ and $v_i(t)$ are the position and velocity of CAV $i$ at time $t$. Constraints (2c), (2d) and (2e) are boundary conditions where $t_i^0$ is the time instant when CAV $i$ enters the Control Zone, $v_i^0$ is the initial speed of CAV $i$, $L$ is the length of the Control Zone segment, and $v_i^f$ is the final speed of CAV $i$. Similar to [6], [8], we assume that the final speeds of all CAVs are the same and fixed, but this assumption can be easily relaxed as shown in [7] and will not influence our analysis on crossing sequences. Constraints (2f) are safety rear-end constraints where CAV $p$ is the CAV physically ahead of CAV $i$, $l$ is the safety distance, and $h_s$ is the safety headway. The form of this constraint is speed-dependent and requires a larger rear-end distance for CAV $i$ with higher speed. However, some related work sets $h_s = 0$ and enlarges the value of $l$ to simplify the calculation. Finally, (2g) and (2h) are physical limitation constraints where $v_{\min,i}$ and $v_{\max,i}$ are the minimum and maximum velocity for CAV $i$, $a_{\min,i}$ and $a_{\max,i}$ are the minimum and maximum acceleration for CAV $i$, respectively. For simplicity, we assume that $v_{\min,i}$, $v_{\max,i}$, $a_{\min,i}$, and $a_{\max,i}$ are the same for all CAVs, and we can handle the situation where these values are dependent on CAV $i$ in the same way. Moreover, in our simulation studies (see Section IV), we will use a more detailed energy model to show that the fuel consumption can be optimized through the simple surrogate model (2).

It is still time-consuming to solve problem (2) through interior point methods or commercial software, e.g., CPLEX. However, due to its simple vehicle dynamics and constraints, we can derive analytical solutions for this problem [6], [7] and quickly obtain the optimal control inputs. It is worth noting that even if the vehicle dynamics and constraints become complicated, we can invoke the Control Barrier Function (CBF) methodology to solve the corresponding optimal control problem efficiently as a sequence of quadratic problems over discretized time instants in $[t_i^0, a_{i,Z_i^1}]$. Interested readers are referred to [18]. For each CAV, we need to solve problem (2) once. As already mentioned, this is done in a *decentralized* way: each CAV solves (2a)-(2h) to determine its own control inputs and the average computation time is smaller than 1 $ms$.

## III. Cooperative Driving Strategies

In this section, we briefly review four cooperative driving strategies used to determine the crossing sequence.

### A. FIFO Strategy and Modified FIFO Strategy

In the FIFO strategy, the crossing sequence follows the FIFO principle. The CAV that enters the Control Zone earlier has a higher crossing priority when a potential conflict with another CAV arises. It is easy to implement this strategy: we only need to add a new incoming CAV at the end of the original crossing sequence and at the same time remove all CAVs that have crossed the Conflict Zone from the crossing sequence.

However, [14] found that this strategy may lead to poor scheduling and possible congestion when the shape of the Control Zone is asymmetrical. We propose a simple idea to overcome this problem, i.e., assign a higher crossing priority to a CAV that is closer to the Conflict Zone. In other words, all CAVs in the Control Zone calculate their distance to the Conflict Zone, and the crossing sequence is derived by sorting CAVs in ascending order in terms of this distance. We call this new strategy the "modified FIFO strategy" and implement it in a time-driven way, i.e., the crossing sequence is periodically updated.

The FIFO strategy and modified FIFO strategy only consider one possible crossing sequence according to their corresponding defining rule. It is easy to see that their time complexities are $O(n)$ and $O(nlog(n))$, respectively, where $n$ is the number of CAVs in the Control Zone. The FIFO strategy is *event-driven*, since it is only invoked whenever a CAV enters the Control Zone or leaves the Conflict Zone so as to update the crossing sequence. The modified FIFO strategy is *time-driven*, since the crossing order is periodically updated based on the current distance of CAVs from the Conflict Zone; in particular, the crossing sequence is updated every $T$ seconds.

### B. Dynamic Resequencing (DR) Strategy

An improvement over strategies based on a single possible crossing sequence is to evaluate several feasible crossing sequences whenever a new CAV enters the Control Zone and to select the optimal one. This is referred to as Dynamic Resequencing. This strategy maintains the relative order of the remaining CAVs and finds an appropriate position in which the new CAV can be inserted so as to optimize a given objective function $\mathcal{J}$. The DR process is shown in **Algorithm 2**. Observe that the DR strategy is implemented in an *event-driven* way with **Algorithm 2** invoked only when the triggering event (a new CAV entering the Control Zone) occurs.

Since the time complexity of computing the objective value of one crossing sequence is $O(n)$ (see step 4 in **Algorithm 2**), the worst time complexity of DR strategy is $O(n^2)$. However, the expected computational complexity is actually $O(Mn)$ where $M$ is the number of lanes. A proof and analysis of the DR strategy and its complexity can be found in [14].

### C. Monte Carlo Tree Search (MCTS) Strategy

As mentioned above, the DR strategy keeps the original crossing order of CAVs unchanged and determines an appropriate insertion position for any new arriving CAV. In contrast, the MCTS-based strategy aims to find the globally optimal crossing sequence among all feasible crossing sequences at every time instant. Clearly, for a real-time implementation it is difficult to enumerate all feasible solutions within a limited computation time, especially when the number of CAVs is large. Thus, this strategy combines a MCTS with some heuristic rules for guiding the search process so as to traverse as many promising crossing sequences as possible.

**Algorithm 2** DR-Based Cooperative Driving Strategy

**Input:** The original crossing sequence $\mathcal{S}$ and the information of all CAVs

**Output:** A new crossing sequence $\mathcal{S}_{new}$

1: Find the preceding vehicle of the new vehicle and its position $k$ in $\mathcal{S}$
2: **for** each $i$ = length($\mathcal{S}$): $-1$: $k$ **do**
3:   Insert the new vehicle into the position $i+1$ of $\mathcal{S}$ and obtain a feasible crossing sequence $\mathcal{S}_f$.
4:   Compute the corresponding objective value $\mathcal{J}_f$ for $\mathcal{S}_f$
5:   **if** $\mathcal{J}_f < \mathcal{J}_{optimal}$ **then**
6:     $\mathcal{J}_{optimal} = \mathcal{J}_f$
7:     $\mathcal{S}_{new} = \mathcal{S}_f$
8:   **end if**
9: **end for**
10: **return** $\mathcal{S}_{new}$

**Algorithm 3** outlines the idea of the MCTS-based strategy; interested readers are referred to [8], [20], [21]. Similar to the modified FIFO strategy, the MCTS is also *time-driven* with the crossing sequence updated every $T$ seconds.

If there is no computation time limit imposed, the time complexity of the MCTS strategy is exponential $O(2^n)$, which is highly undesirable. Nevertheless, the maximum computation time we set can ensure that the search process is finished within an acceptable time dictated by the specific scenario, e.g., 100 $ms$. As validated in [8], the MCTS combined with heuristic rules can always lead to a near-optimal or the optimal crossing sequence even when the search is limited to a very small subset of the search space.

## IV. Simulation Results

In this section, we conduct a series of simulations to compare the performance of the four different strategies outlined in Section III. We assume that vehicle arrivals occur according to Poisson processes (four different ones and one for each entry point) and vary the rate parameters of these Poisson processes to test the performance of the strategies under different traffic demands. In addition, we vary the values of the Control Zone segment lengths to investigate the impact of intersection asymmetries on these strategies.

To accurately describe the arrival of CAVs, we adopt the point-queue model in our simulations [22]. The model assumes

**Algorithm 3** MCTS-Based Cooperative Driving Strategy

**Input:** The information of all CAVs

**Output:** A crossing sequence $\mathcal{S}_{best}$

1: Initialize a root node.
2: **while** the computation budget is not reached **do**
3: Selection: starting at the root node, select the most urgent expandable node based on the UCB1 policy [19].
4: Expansion: randomly select one unvisited child node of the most urgent expandable node to be a new node that is added to the tree.
5: Simulation: run several rollout simulations to determine a complete crossing sequence based on the partial crossing sequence represented by the current new node to evaluate the potential of the new node. Some heuristic rules are utilized to help us quickly capture the real potential of a node during simulation. If the objective value of the crossing sequence obtained from simulation is better than the currently optimal value, record it in $\mathcal{S}_{best}$.
6: Backpropagation: the simulation result is backpropagated through the selected nodes to update the scores of all its parent nodes.
7: **end while**
8: **return** $\mathcal{S}_{best}$

vehicles travel in the free-flow state until they get to the boundary of the intersection we study. To be more concrete, each lane is associated with an independent point-queue. Then, for each lane, we generate the same random number of CAVs generated from a Poisson distribution and let them enter into the point-queue. If the preceding CAV allows adequate space for the first CAV in the point-queue, then this CAV will dequeue and enter the intersection Control Zone. Otherwise, it will stay in the virtual point-queue. In this manner, the actual arrival process of CAVs at each entry point preserves the feasibility constraints (2f) at time $t_i^0$. Thus, the point-queue model has the same effect as the feasibility enforcement zone mechanism described in [23].

In the following comparison, if a strategy is implemented in a time-driven way, we update the crossing sequence every 2 seconds. For each scenario, we simulate a 20-minute traffic process to decrease the influence of random factors. The maximum computation budget for the MCTS strategy is set as 100 $ms$, i.e., the outcome of **Algorithm 3** is used after its execution time reaches this value, unless it has already terminated.

Our performance comparison over the four different strategies is based on three indicators, travel time (delay), energy consumption, and fuel consumption.

1) The travel time (delay) of CAV $i$ is defined as

$$d_i = a_i - \sigma_i, \tag{3}$$

where $a_i$ is the arrival time at the Conflict Zone for CAV $i$, and $\sigma_i$ is the minimum arrival time at the Conflict Zone when CAV $i$ travels at its maximum velocity and acceleration.

2) The energy consumption of CAV $i$ is defined as

$$E_i = \int_{t_i^0}^{a_i} u_i^2(t)dt, \tag{4}$$

where $u_i(t)$ is the control input of vehicle $i$ at time $t$. In actuality, $E_i$ above is only an approximation of a vehicle's energy consumption, since such consumption also depends on speed, and deceleration does not normally contribute to it. However, [24] and [25] pointed out that we can minimize transient engine operation and directly obtain benefits in fuel consumption and emissions by minimizing the $L^2$-norm of the control input. For real-time calculation, we use it as the objective function in problem (2). Besides, we also use a more detailed energy model from [26] to show the actual fuel consumption.

3) The fuel consumption of CAV $i$ from [26] is defined as

$$F_i = \int_{t_i^0}^{a_i} f_{V,i}(t)dt, \tag{5a}$$

$$f_{V,i}(t) = f_{cruise,i}(t) + f_{accel,i}(t), \tag{5b}$$

$$f_{cruise,i}(t) = b_0 + b_1 v_i(t) + b_2 v_i^2(t) + b_3 v_i^3(t), \tag{5c}$$

$$f_{accel,i}(t) = u(t)(c_0 + c_1 v_i(t) + c_2 v_i^2(t)), \tag{5d}$$

where $f_{cruise,i}(t)$ denotes the fuel consumed per second when CAV $i$ drives at a steady velocity $v_i(t)$, and $f_{accel,i}(t)$ is the additional fuel consumed due to the presence of positive acceleration. If $u(t) \leq 0$, $f_{accel,i}(t)$ will be 0 since the engine is rotated by the kinetic energy of the CAV in this case. The unit of fuel consumption is in milliliters ($mL$). $b_0, b_1, b_2, b_3, c_0, c_1$, and $c_2$ are seven model parameters, and here we use the same parameters as in [26], which are obtained through curve-fitting for data from a typical vehicle. Specifically, their values are $b_0 = 0.1569\ mL/s, b1 = 2.450 \times 10^{-2}\ mL/m, b_2 = -7.415 \times 10^{-4}\ mL \cdot s/m^2, b_3 = 5.975 \times 10^{-5}\ mL \cdot s^2/m^3, c_0 = 0.07224\ mL \cdot s/m, c_1 = 9.681 \times 10^{-2}\ mL \cdot s^2/m^2$, and $c_2 = 1.075 \times 10^{-3}\ mL \cdot s^3/m^3$. To distinguish these two energy models, we call the result of $\frac{1}{2}u^2(t)$ "energy consumption" and the result of the above model "fuel consumption" in the following experiments. The experiment results demonstrate that our objective is a simple surrogate function for energy, but the fuel consumption can be optimized by minimizing it.

### A. Comparison Results for Intersection With Same Arrival Rates and Symmetrical Geometry

In this experiment, we set the lengths of all Control Zone segments to be 250 $m$ and the arrival rates at the entry of all lanes at the same value. Then, we vary the arrival rates from 90 $veh/h/lane$ to 420 $veh/h/lane$ to test the performance of the four cooperative driving strategies under different traffic demands. The results are shown in Table I.

It is clear that the MCTS strategy realizes the best traffic efficiency since the average delay is the smallest under all traffic demands, while the DR strategy is the best in terms of

TABLE I
THE COMPARISON RESULTS OF DIFFERENT STRATEGIES FOR THE SYMMETRICAL INTERSECTION

| arrival rates | strategies | average delay ($s$) | average energy consumption | average fuel consumption ($mL$) |
|---|---|---|---|---|
| 90 | MCTS | 2.1770 | 0.0253 | 10.1962 |
| | DR | 2.1770 | 0.0253 | 10.1962 |
| | modified FIFO | 2.2353 | 0.0399 | 10.2275 |
| | FIFO | 2.2353 | 0.0399 | 10.2275 |
| 180 | MCTS | 2.4447 | 0.0990 | 10.3814 |
| | DR | 2.4634 | 0.0849 | 10.3897 |
| | modified FIFO | 2.5822 | 0.1077 | 10.4613 |
| | FIFO | 2.5822 | 0.1077 | 10.4613 |
| 270 | MCTS | 2.6414 | 0.1500 | 10.5293 |
| | DR | 2.7270 | 0.1376 | 10.5728 |
| | modified FIFO | 3.1263 | 0.2586 | 10.8239 |
| | FIFO | 3.1263 | 0.2586 | 10.8239 |
| 360 | MCTS | 3.2677 | 0.4591 | 10.9409 |
| | DR | 3.4248 | 0.3578 | 10.9749 |
| | modified FIFO | 4.4207 | 0.6980 | 11.5847 |
| | FIFO | 4.4145 | 0.6973 | 11.5830 |
| 450 | MCTS | 3.9377 | 0.7258 | 11.3372 |
| | DR | 4.4325 | 0.7664 | 11.5533 |
| | modified FIFO | 6.6261 | 2.0094 | 12.6609 |
| | FIFO | 6.3254 | 1.8600 | 12.5543 |

[1] The arrival rates at the entries of all lanes are the same, and the unit is $veh/h/lane$.
[2] The lengths of all lanes are the same.
[3] The average fuel consumption is the consumption per vehicle per trip.

energy consumption. This reveals a natural trade-off between traffic efficiency and energy consumption. Improving traffic efficiency requires re-planning the crossing sequence and changing the states of CAVs, but this always comes at the cost of additional energy consumption because such re-planning involves frequent acceleration adjustments.

However, when we turn to the fuel consumption calculated from the detailed model, we draw the following two conclusions: 1) the strategy with a much larger average energy consumption will always have a larger fuel consumption than others; 2) when the energy consumption of one strategy is slightly larger than another one, it is hard to say which one is more fuel-intensive since the simplified model ignores the influence of speed while the speed plays an important role in the detailed model. Overall, our results support the approach of optimizing fuel consumption through the simple surrogate model which has the advantage of being computationally much more efficient.

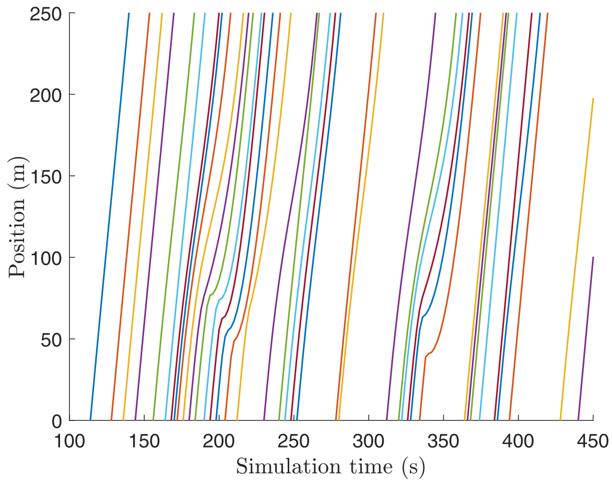
As for the FIFO strategy, one would expect that its energy consumption should be the least since it keeps the order unchanged and CAVs travel according to the control inputs planned when they enter the Control Zone. However, due to the poor coordination performance of the FIFO strategy, especially when the arrival rate is high, there are always many more CAVs in the Control Zone at relatively slow speeds. As a result, CAVs usually need to more frequently decelerate and accelerate compared to other strategies whose coordination performance is better, which leads to much higher energy consumption than the MCTS and DR strategies when running a long simulation.

To validate this intuitive conclusion, Fig. 2 shows partial CAV trajectories on one lane sampled from the FIFO and DR strategy, respectively. During periods $[180, 230]s$ and $[320, 360]s$, we can see visible stop-and-go activities in Fig. 2(a). Due to the safety constraints concerning the arrival time, some CAV slightly decelerates to meet the initial constraints. However, the braking action is continuously amplified and spreads backward. Thus, CAVs brake successively and then accelerate again, which causes a significant added energy consumption. In contrast, the DR strategy allows CAVs to drive more smoothly by adjusting crossing sequences. This demonstrates how improving traffic efficiency by adjusting crossing sequences sometimes indirectly lowers energy consumption by reducing stop-and-go activities, especially when there is a significant gap in traffic efficiency between the two strategies.

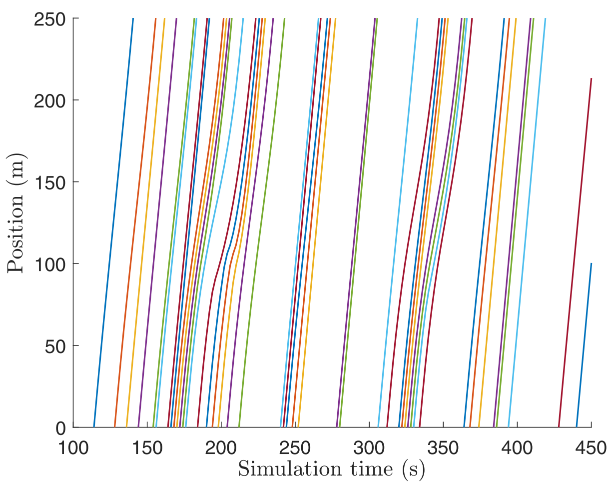
The performance of the modified FIFO strategy and that of the FIFO strategy are approximately the same for this kind of intersection scenario under all arrival rates. Although when the arrival rate is high the orders generated by the two strategies may be a little different, the results are still similar.

### B. Comparison Results for a Geometrically Asymmetrical Intersection

To explore the influence of the geometry of the intersection on the different strategies, we set the length of one lane to

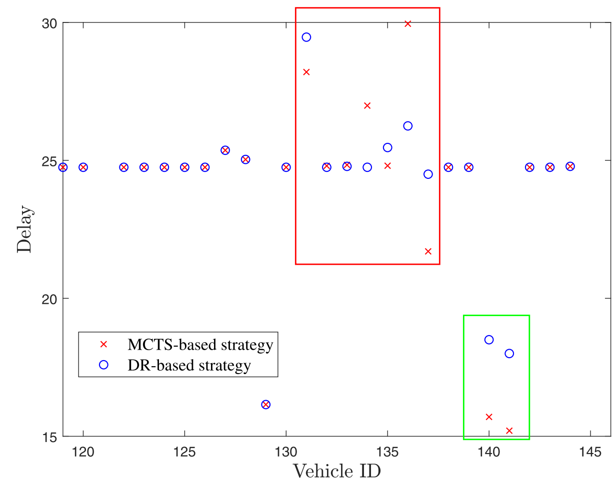



(a) Trajectories from the FIFO-based strategy.

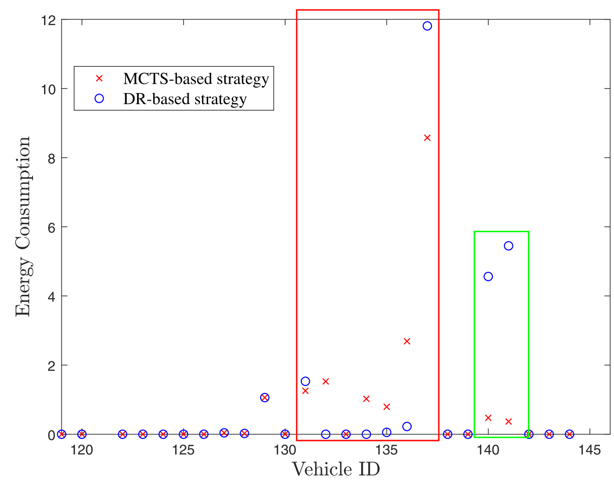



(b) Trajectories from the DR-based strategy.

Fig. 2. Partial vehicle trajectories sampled from different strategies.




(a) Delay of vehicles.




(b) Energy consumption of vehicles

Fig. 3. An example when the MCTS strategy outperforms the DR strategy in all performance metrics.

be 150 $m$ while keeping the lengths of the remaining lanes at 250 $m$. At the same time, we vary the arrival rate from 90 $veh/h/lane$ to 420 $veh/h/lane$ to test the performance of the four cooperative driving strategies under different traffic demands. The results are shown in Table II. Compared to the results shown in Table. I, we can draw two conclusions.

On one hand, the most significant difference is that the modified FIFO strategy shows a much better performance relative to the FIFO strategy. Despite the simplicity of the idea to assign CAVs closer to the Conflict Zone a higher priority - instead of giving CAVs that enter the Control Zone earlier a higher priority, we obtain a significant improvement in performance when the intersection is asymmetrical. However, the performance of the modified FIFO strategy is still unsatisfactory when the arrival rate is high.

On the other hand, the MCTS strategy is the best in terms of traffic efficiency, which is consistent with the findings of the previous experiment. However, our results also show that sometimes both the energy consumption and fuel consumption of the MCTS strategy are superior to that of the DR strategy, e.g., when the arrival rate is 270 $veh/h/lane$. We use a snippet (the results of CAVs whose identity is from 120 to 145 as shown in Fig. 3) from the simulation to explain why this happens.

As we can see from Fig. 3, the crossing sequences generated by the MCTS strategy and the DR strategy for CAV 120 to CAV 130 are the same since their performance is the same. However, they generate different crossing sequences for CAVs 131 through 137, as shown in the red box in Fig. 3(a). Since our primary goal is to decrease the average delay, the MCTS strategy makes a large adjustment to the original crossing sequence by forcing several CAVs ahead of CAV 137 to decelerate so as to allow CAV 137 to pass through the Conflict Zone earlier, hence realizing a small improvement in traffic efficiency. This improvement comes at the cost of higher energy consumption, as can be seen in Fig. 3(b), where the average energy consumption of CAVs under the MCTS strategy is higher than that under the DR strategy. What is interesting to observe is that since the MCTS strategy allows CAV 137 to cross first, the new coming CAVs in the same lane (CAVs 140 and 141) have more ample road space and can access the Conflict Zone with a higher velocity. In contrast, in the DR strategy, these two CAVs are blocked by CAV 137, which results in both the traffic efficiency and energy

TABLE II
THE COMPARISON RESULTS OF DIFFERENT STRATEGIES FOR THE GEOMETRICALLY ASYMMETRICAL INTERSECTION

| arrival rates | strategies | average delay $(s)$ | average energy consumption | average fuel consumption $(mL)$ |
|---|---|---|---|---|
| 90 | MCTS | 2.1934 | 0.0724 | 9.1198 |
| | DR | 2.1934 | 0.0724 | 9.1198 |
| | modified FIFO | 2.2276 | 0.0950 | 9.1487 |
| | FIFO | 3.0406 | 1.0353 | 9.7547 |
| 180 | MCTS | 2.4485 | 0.1927 | 9.4962 |
| | DR | 2.4592 | 0.1817 | 9.4969 |
| | modified FIFO | 2.6281 | 0.2582 | 9.6263 |
| | FIFO | 3.9139 | 1.8221 | 10.4395 |
| 270 | MCTS | 2.6569 | 0.3624 | 9.6725 |
| | DR | 2.7529 | 0.4375 | 9.7182 |
| | modified FIFO | 3.0658 | 0.5392 | 9.9725 |
| | FIFO | 4.6047 | 2.7749 | 10.8996 |
| 360 | MCTS | 3.2419 | 0.7804 | 10.1009 |
| | DR | 3.4250 | 0.9020 | 10.1341 |
| | modified FIFO | 4.5203 | 1.1992 | 10.8750 |
| | FIFO | 5.9534 | 3.7580 | 11.6220 |
| 450 | MCTS | 3.8204 | 2.0426 | 10.4218 |
| | DR | 4.2526 | 1.5560 | 10.4764 |
| | modified FIFO | 6.4420 | 2.3650 | 11.6238 |
| | FIFO | 8.5438 | 5.5766 | 12.6873 |

[1] The arrival rates at the entries of all lanes are the same, and the unit is $veh/h/lane$.
[2] The length of one lane is 150 $m$, while the lengths of remaining lanes are 250 $m$.
[3] The average fuel consumption is the consumption per vehicle per trip.

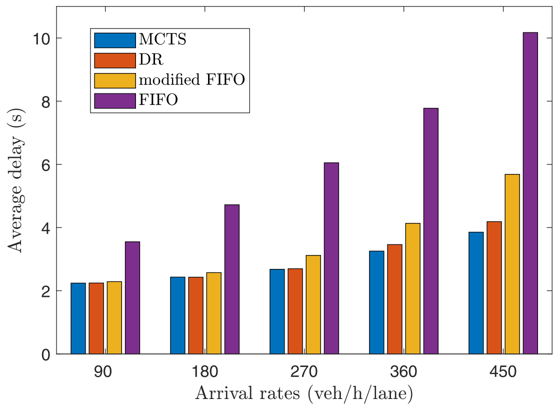
TABLE III
THE COMPARISON RESULTS FOR THE INTERSECTION WITH DIFFERENT ARRIVAL RATES

| arrival rates $(veh/h/lane)$ | strategies | average delay $(s)$ | average energy | average fuel $(mL)$ |
|---|---|---|---|---|
| $\lambda_1 = 180, \lambda_2 = 360,$ $\lambda_3 = 180, \lambda_4 = 360$ | MCTS | 2.5272 | 0.1219 | 10.4142 |
| | DR | 2.5361 | 0.1107 | 10.4117 |
| | modified FIFO | 2.9188 | 0.2076 | 10.6820 |
| | FIFO | 2.9188 | 0.2076 | 10.6820 |
| $\lambda_1 = 225, \lambda_2 = 450,$ $\lambda_3 = 225, \lambda_4 = 450$ | MCTS | 2.9783 | 0.2761 | 10.7485 |
| | DR | 3.0959 | 0.2503 | 10.7973 |
| | modified FIFO | 3.7916 | 0.4511 | 11.2325 |
| | FIFO | 3.7916 | 0.4511 | 11.2325 |
| $\lambda_1 = 180, \lambda_2 = 270,$ $\lambda_3 = 360, \lambda_4 = 450$ | MCTS | 2.7207 | 0.2257 | 10.5892 |
| | DR | 2.7793 | 0.1566 | 10.5764 |
| | modified FIFO | 3.3132 | 0.3017 | 10.9343 |
| | FIFO | 3.3132 | 0.3017 | 10.9343 |
| $\lambda_1 = 270, \lambda_2 = 360,$ $\lambda_3 = 450, \lambda_4 = 540$ | MCTS | 3.2904 | 0.4907 | 10.9975 |
| | DR | 3.4612 | 0.3628 | 10.9978 |
| | modified FIFO | 4.9855 | 0.9128 | 11.8558 |
| | FIFO | 4.9809 | 0.9188 | 11.8574 |

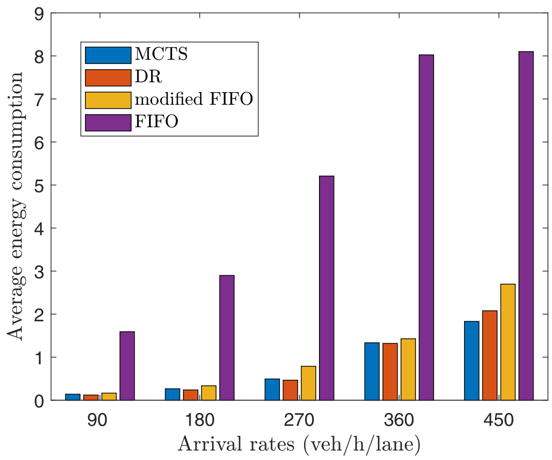
[1] The lengths of all lanes are the same.
[2] The average fuel consumption is the consumption per vehicle per trip.

consumption of the MCTS strategy outperforming that of the DR strategy, as shown in the green boxes in Fig. 3. Of course, this is a consequence of these two CAVs randomly happening to appear. Nonetheless, this example highlights the fact that improving traffic efficiency sometimes indirectly decreases energy consumption.

We have also analyzed the situation where the lengths of lanes in the east-to-west (EW) direction are shorter than

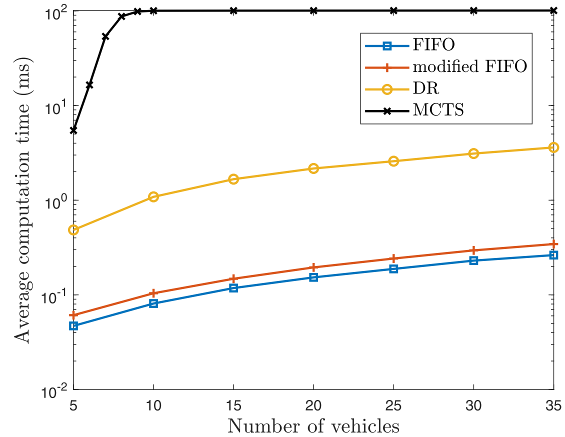



(a) The average delay of different strategies under different arrival rates.

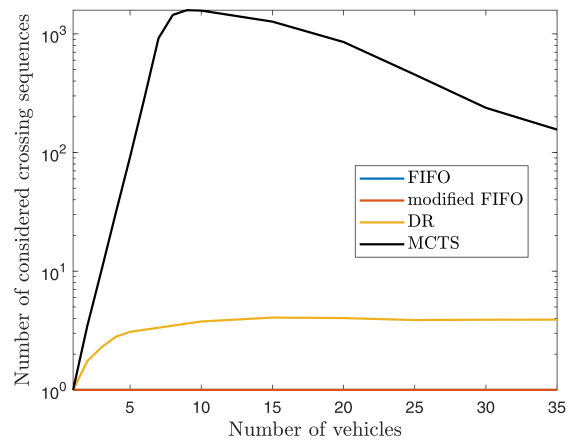



(b) The energy consumption of different strategies under different arrival rates.

Fig. 4. Comparison results of different strategies for the geometrically asymmetrical intersection when the lanes in the EW direction are shorter than that in the NS direction.




(a) The average computation time of different strategies in terms of the number of vehicles in the Control Zone.




(b) The average number of considered crossing sequences of different strategies in terms of the number of vehicles in the Control Zone.

Fig. 5. Comparison results of computation time and crossing sequences.

those in the north-to-south (NS) direction. Specifically, for the intersection shown in Fig. 1, the lengths of Lane 1 and Lane 3 are 150 $m$ while the lengths of Lane 2 and Lane 4 are 250 $m$. The comparison results are shown in Fig. 4, and we can draw similar conclusions as in the above situations.

### C. Comparison Results for Intersection With Different Arrival Rates

In this experiment, we consider a geometrically symmetrical intersection with different arrival rates to investigate the influence of arrival rate asymmetry on different cooperative driving strategies. We denote the arrival rates at the entry points of Lane 1 to Lane 4 as $\lambda_1$, $\lambda_2$, $\lambda_3$, and $\lambda_4$, respectively. The results are shown in Table III where we can see that they are similar to those in Table I, leading to conclusions similar to those drawn from the first experiment. The results also suggest that the asymmetrical arrival rates do not have a noticeable impact on the four strategies, and that the modified FIFO strategy only outperforms the FIFO strategy in geometrically asymmetrical intersections.

### D. Comparison Results in Terms of Computation Time

The computation time is vital for cooperative driving strategies to be applied in practice. In this experiment, we study the computation time every strategy requires to analyze the computation time and the number of crossing sequences they have considered during that time. The results are shown in Fig. 5.

As shown in Fig. 5(a), the computation time increases with the number of vehicles. The FIFO strategy runs the fastest with a computation time of less than 0.3 $ms$ even when there are as many as 35 CAVs in the Control Zone. The computation time of the modified FIFO strategy is only slightly higher. The performance of the DR strategy as a function of CAV numbers is similar, and its computation time is smaller than 4 $ms$. In contrast, the computation time of the MCTS strategy increases exponentially with the number of vehicles until it reaches its assigned maximum computation budget (which was set to 100 $ms$ in this study, which we believe to be small enough for real-time applications). Although we can shorten or prolong the maximum computation time, 100 $ms$ is a value that we have found to strike a good trade-off between performance and computation.

It is of course no wonder that we can consider more crossing sequences with more computation time. As mentioned before, the FIFO strategy and the modified FIFO strategy only consider a single feasible crossing sequence, so their

TABLE IV
THE COMPARISON RESULTS OF DIFFERENT SAFETY REAR-END CONSTRAINTS

| safety constraints | strategies | average delay $(s)$ | average energy consumption | average fuel consumption $(mL)$ |
|---|---|---|---|---|
| $h_s = 0, l = 15m$ | MCTS | 3.3253 | 0.6439 | 11.0614 |
| | DR | 3.5157 | 0.5297 | 11.1144 |
| | modified FIFO | 4.6090 | 1.2347 | 11.7567 |
| | FIFO | 4.5584 | 1.2375 | 11.7473 |
| $h_s = 1, l = 5m$ | MCTS | 3.2981 | 0.6364 | 11.0544 |
| | DR | 3.5197 | 0.7426 | 11.1701 |
| | modified FIFO | 4.6039 | 1.3013 | 11.8197 |
| | FIFO | 4.5664 | 1.4070 | 11.8221 |

computation time performance is similar. The number of crossing sequences that the DR strategy considers always converges to 4 when there are enough vehicles in the Control Zone, a fact consistent with the proof given in [14] that the expected number of crossing sequences the DR strategy considers is equal to the number of lanes (which is 4 in our study). For the MCTS strategy, the number of considered crossing sequences increases exponentially with the number of vehicles at first, since the number of feasible crossing sequences increases exponentially, and there is adequate computation time. Then, when the number of CAVs is larger than 10, the computation time is fixed at 100 $ms$, but the computation time for evaluating a crossing sequence increases with the number of vehicles as the blue and red lines show in Fig. 5(a). Thus, the number of considered crossing sequences starts to decrease. However, we can still search hundreds of feasible crossing sequences even when there is a large number of CAVs in the Control Zone; this ensures the ability of finding a good enough crossing sequence in practice within an acceptable computation time.

### *E. The Analysis on the Safety Rear-End Constraints*

In the above experiments, we set $h_s = 0$ in (2f), so the safety constraints are purely relying on the safety distance $l$. This was made to provide a fair comparison with related work and make the simulation settings similar to other papers. In this experiment, we set $h_s = 1$ to relax this constraint to be speed-dependent and analyze the influence on coordination performance and safety. We should also point out that this assumption was relaxed in a recent paper [27]. We set the arrival rates at the entries of all lanes as 450 $veh/h/lane$. The comparison results of different safety rear-end constraints are shown in Table IV.

The maximum speed is 10 $m/s$, so the minimum safety distances in two types of safety constraints are roughly the same (equal to 15 $m$). Thus, we can see that the results in all metrics are similar.

Then, to further discuss the influence on safety, we use the commonly used safety indicator Time-To-Collision (TTC) [28]:

$$TTC_i(t) = \begin{cases} \dfrac{x_p(t) - x_i(t) - L_p^v}{v_i(t) - v_p(t)}, & \text{if } v_i(t) > v_p(t) \\ \infty, & \text{otherwise} \end{cases} \quad (6)$$

TABLE V
THE PERCENTAGE OF TTC VALUES LOCATED IN SOME INTERVALS

| The percentage of TTC values when $h_s = 0, l = 15m$ | | | | |
|---|---|---|---|---|
| TTC | MCTS | DR | modified FIFO | FIFO |
| $[0, 1]$ | 0.00 | 0.00 | 0.00 | 0.00 |
| $(1, 5]$ | 0.00 | 0.00 | 0.00 | 0.00 |
| $(5, 10]$ | 0.41 | 0.02 | 0.22 | 0.24 |
| $(10, \infty)$ | 99.59 | 99.98 | 99.78 | 99.76 |
| **The percentage of TTC values when $h_s = 1, l = 5m$** | | | | |
| TTC | MCTS | DR | modified FIFO | FIFO |
| $[0, 1]$ | 0.00 | 0.00 | 0.00 | 0.00 |
| $(1, 5]$ | 0.00 | 0.00 | 0.00 | 0.00 |
| $(5, 10]$ | 0.31 | 0.02 | 0.18 | 0.09 |
| $(10, \infty)$ | 99.69 | 99.98 | 99.82 | 99.91 |

where $x_i(t)$ and $v_i(t)$ denote the position and speed of CAV $i$, respectively; $x_p(t)$ and $v_p(t)$ denote the position and speed of the preceding vehicle of CAV $i$, respectively; and $L_p^v$ is the vehicle length of the preceding vehicle.

We record the TTC values of all CAVs at each time point for the above experiment and calculate the percentage of TTC values smaller than 1, 5, and 10 $s$. The statistical results are shown in Table V.

Since all CAVs know their preceding CAVs according to the crossing sequence, they only need to follow them with a steady velocity and a suitable distance. The difference between $v_i(t)$ and $v_p(t)$ is small, so $TTC_i(t)$ is usually a large value. We can see that, in this case, no TTC value is smaller than 5 $s$, indicating that safety is guaranteed. After we vary arrival rates and test many other cases, we find that nearly no TTC value lies in the interval [0, 1], and no collision ever happens. Thus, our safety constraints can ensure the safety of CAVs.

## V. THE TRADE-OFF BETWEEN FAIRNESS AND EFFICIENCY

In this section, we explore the question regarding why resequencing still provides benefits in heavy traffic. One might expect that in situations where the intersection is congested, there would be little or no flexibility for improving performance. After analyzing a large number of simulations in our study, we believe that there are mainly two reasons for this

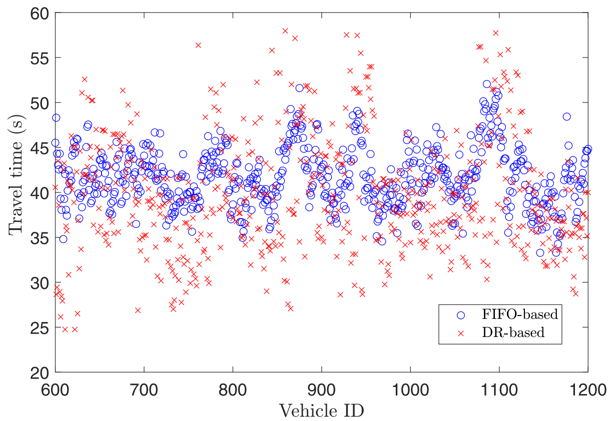



Fig. 6. Distribution of CAV travel times under two different cooperative strategies.

phenomenon. The first one is the subzone division of the Conflict Zone, and the second is that the traffic efficiency is often improved at the cost of causing *unfairness* in extreme traffic situations.

Referring to Fig. 1, suppose that CAV 1 in Lane 1 goes straight, CAV 2 in Lane 2 turns left, and CAV 3 in Lane 3 goes straight, and their distances to the Conflict Zone are similar. Then, it is easy to prove that crossing sequence 132 is better than 123 since CAV 1 and CAV 3 can pass through the intersection at the same time. Thus, even when traffic is congested, we can still improve traffic efficiency by pairing non-conflicting CAVs through resequencing.

When the total arrival rate at all lanes $\lambda = \sum_{i=1}^{n} \lambda_i$ is very close to or larger than the maximum arrival rate $\lambda_{\max}$ that the intersection can handle (i.e., its traffic capacity), no control strategy can alleviate congestion effectively. However, we find that strategies with and without resequencing behave differently in this extreme situation. Suppose that the arrival rates at all lanes are the same, i.e., $\lambda_1 = \lambda_2 = \lambda_3 = \lambda_4$ and the queue lengths of all lanes are the same, i.e., $q_1 = q_2 = q_3 = q_4$. Then, due to the same arrival rate, CAVs roughly arrive at all lanes evenly, which leads to CAVs in each lane passing through the Conflict Zone in turn under the strategies without resequencing, e.g., the FIFO strategy. However, strategies with resequencing try to insert a new arriving CAV into a front position which provides better performance. Sometimes, due to pairings and the randomness of the traffic process, CAVs in one lane may leave faster than other lanes. Then, a new arriving CAV at this lane has a much higher probability of finding a CAV to pair with and a better position in the current crossing sequence. The congestion in this lane may gradually dissipate while the congestion in other lanes builds up. Thus, we conclude that strategies with resequencing tend to block one or several lanes and allow CAVs in the remaining lanes to pair up with non-conflicting CAVs near the Conflict Zone and pass through the intersection quickly. Since this kind of strategy can increase the number of vehicle pairs, it improves traffic efficiency.

We use an experiment to validate this idea where we set the total arrival rate of this intersection to a very large value ($\lambda = 2\ veh/s$ and $\lambda_i = 0.5\ veh/s$, $i = \{1, 2, 3, 4\}$), much larger than the maximum arrival rate. We also set the safety time-headway for right-turn and going straight as $1.5s$ and for left-turn as $2.5\ s$. For simplicity, the following analysis assumes the safety time-headway for all actions is $1.5\ s$ (the service time of the intersection is $1.5\ s$). Then, in the ideal situation, we can have two CAVs passing through the Conflict Zone at the same time to maximize the utilization of the road resources (that is, the intersection can serve two vehicles at one time); clearly, the actual efficiency is lower than this. Then, in this ideal situation, the minimum headway (service time) is $1.5/2 = 0.75\ s$. The maximum arrival rates should be $1/0.75 = 4/3\ veh/s$. If the arrival rate is larger than this value, the number of vehicles will be larger than the capacity of the intersection leading to traffic congestion. In this case, the travel times of CAVs under the FIFO strategy and the DR strategy are shown in Fig. 6.

It is clear that the travel times of CAVs under the FIFO strategy have a much lower variance than those under the DR strategy. In particular, the mean travel time and standard deviation under the DR strategy are 39.06s and 6.65s, respectively, while under the FIFO strategy they are 42.27s and 3.41s. This example shows that traffic efficiency (lower mean travel time) may come at the cost of bigger standard deviation and creates a natural trade-off problem. To give a better quantitative analysis, we first define an "unfairness" metric $\rho$ as the standard deviation of the arrival times,

$$\rho = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(a_i - \mu)^2}, \tag{7}$$

$$\mu = \frac{1}{N}\sum_{i=1}^{N} a_i, \tag{8}$$

where $N$ is the number of all CAVs, $a_i$ is the arrival time at the Conflict Zone for CAV $i$, and $\mu$ is the average travel time of all CAVs. A larger $\rho$ means more significant unfairness.

We point out, however, that this problem typically arises at high traffic rates, since resequencing is beneficial to *all* CAVs when the arrival rate is not too high; at high traffic rates, however, resequencing can only improve the overall traffic efficiency by sacrificing the performance of *some* CAVs due to the limited road resources to be allocated.

A simple method of balancing performance when the DR strategy is used is based on introducing a balancing factor $\alpha$. In particular, we only adjust the crossing sequence when the following condition is satisfied:

$$J_{new} < J_{best} - \alpha J_{new}, \tag{9}$$

where $J_{new}$ is the objective value of the new crossing sequence and $J_{best}$ is the currently optimal objective value. In other words, there is an incentive to update the crossing sequence only when the performance of the new crossing sequence is much better than the original one. We vary the value of $\alpha$ from 0 to 3% to show the trade-off between efficiency and fairness in Fig. 7.

Looking at Fig. 7, it is clear that when we increase the value of $\alpha$ we improve fairness (lower travel time standard deviation) by decreasing efficiency, i.e., a larger $\alpha$ implies greater emphasis on fairness. We observe that there may be a

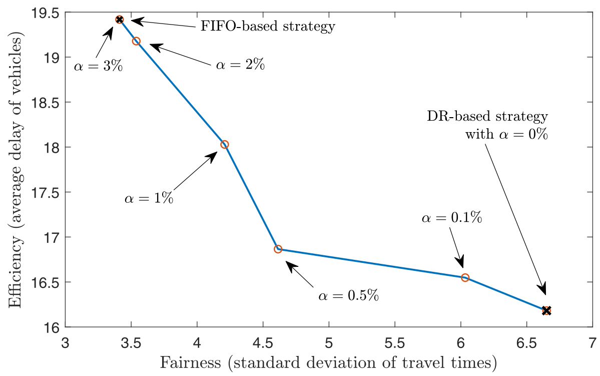



Fig. 7. Trade-off between efficiency and fairness under different balancing factors $\alpha$.

Pareto optimal point (the point with $\alpha = 0.5\%$ in Fig. 7) that achieves a balance between the two criteria: a perturbation to its left results in significant efficiency relative to fairness, with the situation reversed for perturbations to its right. This paves the way for future research in this interesting direction.

Along similar lines, for the MCTS strategy we can also make some modifications to consider fairness in the search process. In the original MCTS strategy, we use the following UCB1 policy to determine the most urgent expandable node:

$$\arg\max_i \; Q_i + C\sqrt{\frac{\ln n}{n_i}}, \tag{10}$$

where $Q_i$ is the score of child node $i$ and $Q_i \in [0, 1]$. Moreover, $n$ is the number of times the current node has been visited, $n_i$ is the number of times child node $i$ has been visited, and $C$ is a weight parameter. The child node with the largest total score is selected. The objective is to prevent significant change in order with a resulting small benefit. Thus, we propose to add a penalty term $P_i$ to the original UCB1 policy defined as

$$P_i = \beta D_i, \tag{11}$$

where $D_i$ is an integer indicating how many orders are different between the new crossing sequence and a reference crossing sequence, e.g., the currently optimal crossing sequence or the desired crossing sequence, and $\beta$ is a negative weight for penalizing the difference. Then, the modified UCB1 policy is

$$\arg\max_i \; Q_i + P_i + C\sqrt{\frac{\ln n}{n_i}}. \tag{12}$$

Using this policy, the MCTS only explores significantly different crossing sequences when it finds that such crossing sequences can bring a much improved traffic efficiency. Note that we can also consider energy or other metrics in the objective or modify the heuristic rules involved according to the desired performance priorities.

## VI. Conclusion and Future Research

This paper compares the performance of some state-of-the-art cooperative driving strategies under various influencing factors, including symmetrical intersections, asymmetrical intersections, and asymmetrical arrival rates. Our main conclusion is that the MCTS and DR strategies both perform well in all scenarios and are recommended for use in practice. However, we have also pointed out that efficiency sometimes comes at the cost of fairness to a certain subset of CAVs. Through some modifications to these strategies, we have shown how to control the trade-off between fairness and efficiency.

Although we have only considered an intersection with a single lane in each direction, the conclusions of this study can be extended to other driving scenarios, e.g., highway on-ramps and intersections with multiple lanes. There are many problems deserve to be studied deeply, e.g., how to accelerate the search process of the MCTS strategy further, how to choose a proper reference crossing sequence, and so on. Due to space limitations, we will omit here and leave it for the future.

## References

[1] J. Rios-Torres and A. A. Malikopoulos, "A survey on the coordination of connected and automated vehicles at intersections and merging at highway on-ramps," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 5, pp. 1066–1077, May 2017.

[2] L. Chen and C. Englund, "Cooperative intersection management: A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 2, pp. 570–586, Feb. 2016.

[3] L. Li, D. Wen, and D. Yao, "A survey of traffic control with vehicular communications," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 1, pp. 425–432, Feb. 2014.

[4] K. Dresner and P. Stone, "Multiagent traffic management: A reservation-based intersection control mechanism," in *Proc. 3rd Int. Joint Conf. Auton. Agents Multiagent Syst.*, vol. 2, Jul. 2004, pp. 530–537.

[5] K. Dresner and P. Stone, "A multiagent approach to autonomous intersection management," *J. Artif. Intell. Res.*, vol. 31, pp. 591–656, Mar. 2008.

[6] 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, Jul. 2018.

[7] 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, pp. 1–9, Nov. 2019.

[8] H. Xu, Y. Zhang, L. Li, and W. Li, "Cooperative driving at unsignalized intersections using tree search," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 11, pp. 4563–4571, Nov. 2020.

[9] P. T. Li and X. 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," *Transp. Res. B, Methodol.*, vol. 105, pp. 479–506, Nov. 2017.

[10] S. A. Fayazi and A. Vahidi, "Mixed-integer linear programming for optimal scheduling of autonomous vehicle intersection crossing," *IEEE Trans. Intell. Veh.*, vol. 3, no. 3, pp. 287–299, Sep. 2018.

[11] L. Li and F.-Y. Wang, "Cooperative driving at blind crossings using intervehicle communication," *IEEE Trans. Veh. Technol.*, vol. 55, no. 6, pp. 1712–1724, Nov. 2006.

[12] H. Xu, S. Feng, Y. Zhang, and L. Li, "A grouping-based cooperative driving strategy for CAVs merging problems," *IEEE Trans. Veh. Technol.*, vol. 68, no. 6, pp. 6125–6136, Jun. 2019.

[13] Y. Meng, L. Li, F.-Y. Wang, K. Li, and Z. Li, "Analysis of cooperative driving strategies for nonsignalized intersections," *IEEE Trans. Veh. Technol.*, vol. 67, no. 4, pp. 2900–2911, Apr. 2018.

[14] Y. Zhang and C. G. Cassandras, "A decentralized optimal control framework for connected automated vehicles at urban intersections with dynamic resequencing," in *Proc. IEEE Conf. Decis. Control (CDC)*, Dec. 2018, pp. 217–222.

[15] M. Choi, A. Rubenecia, and H. H. Choi, "Reservation-based cooperative traffic management at an intersection of multi-lane roads," in *Proc. Int. Conf. Inf. Netw. (ICOIN)*, Jan. 2018, pp. 456–460.

[16] R. Hult, G. R. Campos, E. Steinmetz, L. Hammarstrand, P. Falcone, and H. Wymeersch, "Coordination of cooperative autonomous vehicles: Toward safer and more efficient road transportation," *IEEE Signal Process. Mag.*, vol. 33, no. 6, pp. 74–84, Nov. 2016.

[17] X. Qian, J. Gregoire, A. de La Fortelle, and F. Moutarde, "Decentralized model predictive control for smooth coordination of automated vehicles at intersection," in *Proc. Eur. Control Conf. (ECC)*, Jul. 2015, pp. 3452–3458.

[18] W. Xiao, C. G. Cassandras, and C. Belta, "Decentralized merging control in traffic networks with noisy vehicle dynamics: A joint optimal control and barrier function approach," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, Oct. 2019, pp. 3162–3167.

[19] L. Kocsis and C. Szepesvári, "Bandit based Monte-Carlo planning," in *Proc. Eur. Conf. Mach. Learn.* Springer, 2006, pp. 282–293.

[20] C. B. Browne *et al.*, "A survey of Monte Carlo tree search methods," *IEEE Trans. Comput. Intell.*, vol. 4, no. 1, pp. 1–43, Mar. 2012.

[21] H. Xu, Y. Zhang, C. G. Cassandras, L. Li, and S. Feng, "A bi-level cooperative driving strategy allowing lane changes," *Transp. Res. C, Emerg. Technol.*, vol. 120, Nov. 2020, Art. no. 102773.

[22] X. J. Ban, J. S. Pang, H. X. Liu, and R. Ma, "Continuous-time point-queue models in dynamic network loading," *Transp. Res. B, Methodol.*, vol. 46, no. 3, pp. 360–380, Mar. 2012.

[23] Y. Zhang, C. G. Cassandras, and A. A. Malikopoulos, "Optimal control of connected automated vehicles at urban traffic intersections: A feasibility enforcement analysis," in *Proc. Amer. Control Conf. (ACC)*, May 2017, pp. 3548–3553.

[24] A. A. Malikopoulos, "Stochastic optimal control for series hybrid electric vehicles," in *Proc. Amer. Control Conf.*, Jun. 2013, pp. 1189–1194.

[25] J. Rios-Torres and A. A. Malikopoulos, "Automated and cooperative vehicle merging at highway on-ramps," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 4, pp. 780–789, Apr. 2017.

[26] M. A. S. Kamal, M. Mukai, J. Murata, and T. Kawabe, "Model predictive control of vehicles on urban roads for improved fuel economy," *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 3, pp. 831–841, May 2013.

[27] Y. Zhang and C. G. Cassandras, "Joint time and energy-optimal control of connected automated vehicles at signal-free intersections with speed-dependent safety guarantees," in *Proc. IEEE 58th Conf. Decis. Control (CDC)*, Dec. 2019, pp. 329–334.

[28] K. Vogel, "A comparison of headway and time to collision as safety indicators," *Accident Anal. Prevention*, vol. 35, no. 3, pp. 427–433, May 2003.


**Huile Xu** received the B.S. degree from Chongqing University, China, in 2016. He is currently pursuing the Ph.D. degree with the Department of Automation, Tsinghua University, China. He is also a Visiting Ph.D. Student with the Division of Systems Engineering, Boston University, Boston, and the Center for Information and Systems Engineering, Boston University. His research interests include cooperative driving, intelligent vehicles, and optimal control.

**Christos G. Cassandras** (Life Fellow, IEEE) received the degrees from Yale University, Stanford University, and Harvard University. From 1982 to 1984, he was with ITP Boston Inc., where he worked on the design of automated manufacturing systems. In 1984 to 1996, he was a Faculty Member with the Department of Electrical and Computer Engineering, University of Massachusetts/Amherst. He has most recently collaborated with MathWorks Inc., in the development of the discrete event and hybrid system simulator SimEvents. He is currently a Distinguished Professor of engineering with Boston University. He is also the Head of the Division of Systems Engineering, a Professor of electrical and computer engineering, and the Co-Founder of the Center for Information and Systems Engineering (CISE), Boston University. In addition to his academic activities, he has worked extensively with industrial organizations on various systems integration projects and the development of decision support software. He specializes in the areas of discrete event and hybrid systems, cooperative control, stochastic optimization, and computer simulation, with applications to computer and sensor networks, manufacturing systems, and transportation systems. He has published over 400 refereed articles in these areas and six books. He is a fellow of the IFAC. He is also a member of the Phi Beta Kappa and the Tau Beta Pi. He was the 2012 President of the IEEE Control Systems Society (CSS). He has also served as the Vice President for Publications and on the Board of Governors of the CSS, and on several IEEE committees, and has chaired several conferences. He was a recipient of several awards, including the 2011 IEEE Control Systems Technology Award, the Distinguished Member Award from the IEEE Control Systems Society in 2006, the 1999 Harold Chestnut Prize (IFAC Best Control Engineering Textbook) for *Discrete Event Systems: Modeling and Performance Analysis*, the 2011 Prize and the 2014 Prize for the IBM/IEEE Smarter Planet Challenge Competition, the 2014 Engineering Distinguished Scholar Award at Boston University, several honorary professorships, the 1991 Lilly Fellowship and the 2012 Kern Fellowship. He has been a Plenary/a Keynote Speaker at numerous international conferences, including the 2017 IFAC World Congress, the American Control Conference in 2001 and the IEEE Conference on Decision and Control in 2002 and 2016. He was the Editor-in-Chief of the IEEE TRANSACTIONS ON AUTOMATIC CONTROL from 1998 to 2009, and has also served as an Editor for Technical Notes and Correspondence and an Associate Editor. He has guest-edited several technical journal issues and currently serves on several journal Editorial Boards, including an Editor of *Automatica*. He has also been an IEEE Distinguished Lecturer.

**Li Li** (Fellow, IEEE) is currently an Associate Professor with the Department of Automation, Tsinghua University, Beijing, China, where he is involved in the fields of complex and networked systems, intelligent control and sensing, intelligent transportation systems, and intelligent vehicles. He has published over 90 SCI indexed international journal articles and over 70 international conference papers as a First/Corresponding author. He serves as an Associate Editor for the IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, a member for the Editorial Advisory Board for *Transportation Research Part C: Emerging Technologies*, and the Editorial Board for *Transport Reviews*.

**Yi Zhang** (Member, IEEE) received the B.S. and M.S. degrees from Tsinghua University, China, in 1986 and 1988, respectively, and the Ph.D. degree from the University of Strathclyde, U.K., in 1995. He is currently a Professor in control science and engineering with Tsinghua University. His current research interests focusing on intelligent transportation systems. His active research areas include intelligent vehicle-infrastructure cooperative systems, analysis of urban transportation systems, urban road network management, traffic data fusion and dissemination, and urban traffic control and management. His research fields also cover the advanced control theory and applications, advanced detection and measurement, and systems engineering.