

A Priority-Aware Replanning and Resequencing Framework for Coordination of Connected and Automated Vehicles

Behdad Chalaki¹, *Student Member, IEEE*, and Andreas A. Malikopoulos², *Senior Member, IEEE*

Abstract—Deriving optimal control strategies for coordination of connected and automated vehicles (CAVs) often requires re-evaluating the strategies in order to respond to unexpected changes in the presence of disturbances and uncertainties. In this letter, we first extend a decentralized framework that we developed earlier for coordination of CAVs at a signal-free intersection to incorporate replanning. Then, we further enhance the framework by introducing a priority-aware resequencing mechanism which designates the order of decision making of CAVs based on theory from the job-shop scheduling problem. Our enhanced framework relaxes the first-come-first-serve decision order which has been used extensively in these problems. We illustrate the effectiveness of our proposed approach through numerical simulations.

Index Terms—Connected and automated vehicles, replanning, resequencing, sequential decision making.

I. INTRODUCTION

SEVERAL research efforts in the literature have considered a two-level optimization framework for coordination of connected and automated vehicles (CAVs) at traffic bottlenecks. An *upper-level* optimization yields, for each CAV, the optimal time to exit the control zone, while a *low-level* optimization yields for the CAV its optimal control input (acceleration/deceleration) to achieve the optimal time derived in the upper-level subject to the state, control, and safety constraints. There have been several approaches in the literature to solve the upper-level optimization problem, including first-in-first-out (FIFO) queuing policy [1], [2], heuristic Monte Carlo tree search methods [3], [4], centralized optimization techniques [5], [6], and job-shop scheduling [7], [8]. Given the solution of the upper-level optimization problem, the constrained optimal control problem is solved sequentially in

the low-level optimization, yielding the optimal control input for each CAV. To solve the low-level optimization problem, research efforts have used optimal control techniques to derive the closed form solutions [1], [9]–[11], or model predictive control (MPC) [6], [12]–[14].

To the best of our knowledge, there have been limited studies in exploring the effects of decision-making sequence in the low-level optimization problem. de Campos *et al.* [15] presented a heuristic approach to find a decision order for CAVs at an intersection based on their time to reach an unsafe set, i.e., the CAV can no longer stop before the intersection. Alrifaae *et al.* [16] proposed a graph-based approach to construct levels of parallelizable agents for non-cooperative decentralized MPC, in which agents on the same level solve the problem in parallel, sequentially after agents on the previous level. Xiao and Cassandras [17] relaxed the FIFO queuing policy by formulating a resequencing problem before the control zone. The authors assumed that after a CAV performs the resequencing, its speed remains constant until it arrives at the control zone.

In this letter, we build upon the framework introduced in [10] consisting of a single optimization level aimed at both minimizing energy consumption and improving the traffic throughput. Using the proposed framework, each CAV computes the optimal exit time corresponding to an unconstrained energy optimal trajectory which satisfies all the state, control, and safety constraints. We extend this letter by integrating the replanning mechanism into the framework. Since unexpected changes in the presence of disturbances and uncertainties can result in deviations from the optimal planned trajectory of the CAVs, the replanning mechanism introduces feedback in the planning which can respond to these changes in the system to some extent. In addition, using the theory from the job-shop scheduling problem, we further enhance the framework by introducing a priority-aware resequencing mechanism to find the decision making sequence of the CAVs based on the minimum exit time from the traffic bottleneck.

The work that we report on this letter advances the state of the art in a way that relaxes the first-come-first-serve (FCFS) decision making sequence of the CAVs. The contributions of this letter are: (i) the introduction of replanning as a feedback mechanism to handle uncertainties or disturbances, and (ii) the

Manuscript received September 12, 2021; revised November 16, 2021; accepted December 2, 2021. Date of publication December 7, 2021; date of current version December 15, 2021. This work was supported by ARPAE's NEXTCAR Program under Award DE-AR0000796. Recommended by Senior Editor L. Menini. (*Corresponding author: Behdad Chalaki.*)

The authors are with the Department of Mechanical Engineering, University of Delaware, Newark, DE 19716 USA (e-mail: bchalaki@udel.edu; andreas@udel.edu).

Digital Object Identifier 10.1109/LCSYS.2021.3133416

2475-1456 © 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See <https://www.ieee.org/publications/rights/index.html> for more information.

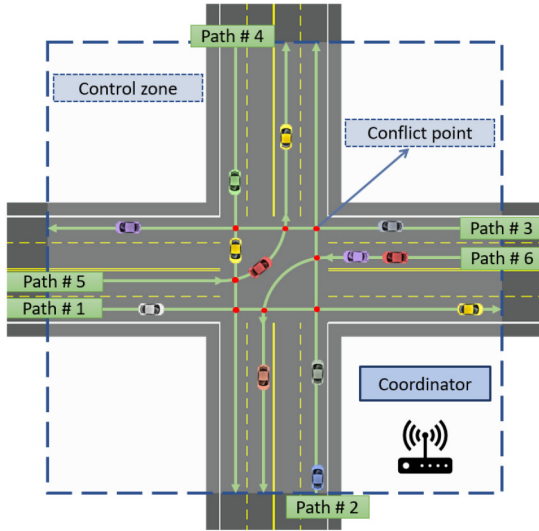


Fig. 1. A signal free intersection with conflict points.

development of a priority-aware resequencing mechanism for the coordination of CAVs.

The remainder of this letter is structured as follows. In Section II, we introduce the modeling framework and we present the priority-aware resequencing mechanism in Section III. Finally, we provide simulation results in Section IV, and concluding remarks in Section V.

II. MODELING FRAMEWORK

We consider a signal-free intersection (Fig. 1), which includes a *coordinator* that stores information about the intersection's geometry and CAVs' trajectories. The coordinator acts as a database for the CAVs and does not make any decision. The intersection includes a *control zone* inside of which the CAVs can communicate with the coordinator. We call the points inside the control zone where paths of CAVs intersect and a lateral collision may occur as *conflict points*. Let $\mathcal{O} \subset \mathbb{N}$ index the set of conflict points, $N(t) \in \mathbb{N}$ be the total number of CAVs inside the control zone at time $t \in \mathbb{R}_{\geq 0}$, and $\mathcal{N}(t) = \{1, \dots, N(t)\}$ be the queue that designates the order in which each CAV entered the control zone. We model the dynamics of each CAV $i \in \mathcal{N}(t)$ as a double integrator

$$\begin{aligned} \dot{p}_i(t) &= v_i(t), \\ \dot{v}_i(t) &= u_i(t), \end{aligned} \quad (1)$$

where $p_i(t) \in \mathcal{P}_i$, $v_i(t) \in \mathcal{V}_i$, and $u_i(t) \in \mathcal{U}_i$ denote position, speed, and control input at t , respectively. The sets \mathcal{P}_i , \mathcal{V}_i , and \mathcal{U}_i , for $i \in \mathcal{N}(t)$, are compact subsets of \mathbb{R} . Let $t_i^0 \in \mathbb{R}_{\geq 0}$ be the time that CAV $i \in \mathcal{N}(t)$ enters the control zone, and $t_i^f > t_i^0 \in \mathbb{R}_{\geq 0}$ be the time that CAV i exits the control zone. For each CAV $i \in \mathcal{N}(t)$, the control input and speed are bounded by

$$u_{i,\min} \leq u_i(t) \leq u_{i,\max}, \quad (2)$$

$$0 < v_{\min} \leq v_i(t) \leq v_{\max}, \quad (3)$$

where $u_{i,\min}$, $u_{i,\max}$ are the minimum and maximum control inputs and v_{\min} , v_{\max} are the minimum and maximum speed limits, respectively.

We consider that CAVs do not perform any lane-change maneuver, and thus there are finite paths among which they can choose. The set of all possible paths in the control zone is given by $\mathcal{L} = \{1, \dots, z\}$, $z \in \mathbb{N}$. The path of the CAV $i \in \mathcal{N}(t)$ in the control zone is denoted by $\ell_i \in \mathcal{L}$ (Fig. 1), and is decided a priori based on some upper-level routing problem. In our modeling framework, we make the following assumption.

Assumption 1: Each path of a CAV cannot get either split to two paths or merged by another CAV's path.

This assumption implies that there should be separate lanes for the turning maneuvers at the intersections. This might be a strong assumption, but it simplifies the complexity of our resequencing algorithm (formally defined next) by only considering the rear-end safety constraints for the CAVs travelling on the same path. Investigating the implications of relaxing this assumption is the subject of ongoing research.

To guarantee rear-end safety between CAV $i \in \mathcal{N}(t)$ and a preceding CAV $k \in \mathcal{N}(t) \setminus \{i\}$, we impose the following constraint,

$$p_k(t) - p_i(t) \geq \delta_i(t) = \gamma + \varphi \cdot v_i(t), \quad (4)$$

where $\delta_i(t)$ is the safe speed-dependent distance, while γ and $\varphi \in \mathbb{R}_{>0}$ are the standstill distance and reaction time, respectively.

Let CAV $k \in \mathcal{N}(t) \setminus \{i\}$ be a CAV that has already planned its trajectory which might cause a lateral collision with CAV i . We denote by p_i^n and p_k^n the distance of the conflict point $n \in \mathcal{O}$ from i 's and k 's paths' entries, respectively. Since we do not use the FIFO queuing policy, CAV i can reach at conflict point n either after or before CAV k . In the first case, i.e., when CAV i reaches at conflict point n after CAV k , we have

$$p_i^n - p_i(t) \geq \delta_i(t), \quad \forall t \in [t_i^0, t_i^n], \quad (5)$$

where t_i^n is the known time that CAV k reaches at conflict point n , i.e., position p_k^n . In the second case, where CAV i reaches at the conflict point n before CAV k , we have

$$p_k^n - p_k(t) \geq \delta_k(t) = \gamma + \varphi \cdot v_k(t), \quad \forall t \in [t_k^0, t_i^n], \quad (6)$$

where t_i^n is determined by the trajectory planned by CAV i . Since $0 < v_{\min} \leq v_i(t)$, the position $p_i(t)$ is a strictly increasing function. Thus, the inverse $t_i(\cdot) = p_i^{-1}(\cdot)$ exists and it is called the *time trajectory* of CAV i [10]. Hence, we have $t_i^n = p_i^{-1}(p_i^n)$. Therefore, for each candidate path of CAV i , there exists a unique time trajectory which can be evaluated at conflict point n , i.e., $t_i(p_i^n)$, to find the time that CAV i reaches at conflict point n , i.e., t_i^n .

By moving all terms in (5) to the RHS, we get $\delta_i(t) + p_i(t) - p_i^n \leq 0$. Constraint (5) is satisfied, if $\max(\delta_i(t) + p_i(t) - p_i^n) \leq 0$ in the interval $[t_i^0, t_i^n]$. Likewise, if $\max(\delta_k(t) + p_k(t) - p_k^n) \leq 0$ in the interval $[t_k^0, t_i^n]$ constraint (6) is satisfied. However, to ensure the lateral safety between CAV i and CAV k at conflict point n , either (5) or (6) must be satisfied, and thus we impose the lateral safety constraint on CAV i using minimum

function as

$$\min \left\{ \begin{array}{l} \max_{t \in [t_i^0, t_k^0]} \{\delta_i(t) + p_i(t) - p_i^n\}, \\ \max_{t \in [t_k^0, t_i^n]} \{\delta_k(t) + p_k(t) - p_k^n\} \end{array} \right\} \leq 0. \quad (7)$$

In our framework, each CAV i communicates with the coordinator to solve a time minimization problem, which determines t_i^f , i.e., the time that CAV i must exit the control zone. The time t_i^f corresponds to the unconstrained energy optimal trajectory guaranteeing that state, control, and safety constraints are satisfied. This trajectory is communicated back to the coordinator, so that the subsequent CAVs receive this information and plan their trajectories accordingly. Our framework implies that the CAVs do not have to come to a full stop at the intersection, thereby conserving momentum and fuel while also improving travel time. By enforcing the unconstrained energy-optimal trajectory that guarantees the satisfaction of all the state, control, and safety constraints, we avoid inherent real-time implementation difficulties in solving a constrained optimal control and piecing constrained and unconstrained arcs together [10], [18].

We start our exposition with the unconstrained energy optimal solution of CAV i , which has the following form [1]

$$\begin{aligned} u_i(t) &= 6a_it + 2b_i, \\ v_i(t) &= 3a_it^2 + 2b_it + c_i, \\ p_i(t) &= a_it^3 + b_it^2 + c_it + d_i, \end{aligned} \quad (8)$$

where a_i, b_i, c_i, d_i are constants of integration. CAV i must also satisfy the boundary conditions

$$p_i(t_i^0) = 0, \quad v_i(t_i^0) = v_i^0, \quad (9)$$

$$p_i(t_i^f) = p_i^f, \quad u_i(t_i^f) = 0, \quad (10)$$

where $u_i(t_i^f) = 0$ because the speed at the exit of the control zone is not specified [19]. The details of the derivation of the unconstrained solution are discussed in [1].

One of the advantages of incorporating replanning in the framework is introducing feedback in the system. Replanning can occur either periodically (i.e., at a period determined a priori) or be event-driven (i.e., based on an occurrence of a certain event such as the entrance of a new CAV in the control zone). All CAVs in the control zone observe their state at each replanning instance and re-solve their optimization problem, discussed next, sequentially with the new initial conditions. For CAV i , let $\tau \in [t_i^0, t_i^f]$ be the replanning time, and $\tilde{\mathbf{x}}_i(\tau) = [\tilde{p}_i(\tau) \tilde{v}_i(\tau)]^\top$, be the measurement of the state at this time. The revised initial conditions for CAV i at this replanning instance is given by

$$p_i(\tau) = \tilde{p}_i(\tau), \quad v_i(\tau) = \tilde{v}_i(\tau). \quad (11)$$

Definition 1: The compact set $\mathcal{T}_i(\tau) = [t_i^{f,\tau}, \bar{t}_i^{f,\tau}]$ is the set of feasible solution of CAV $i \in \mathcal{N}(t)$ for the exit time, where $t_i^{f,\tau}$ and $\bar{t}_i^{f,\tau}$ denotes the minimum and maximum feasible exit time computed at τ . CAV i can determine $\mathcal{T}_i(\tau)$ at time τ

using the speed and control input constraints (2)-(3), initial condition (9) or (11) (depending on if $\tau = t_i^0$), and final condition (10). For the derivation of this compact set, refer to [18].

To avoid abrupt changes in the control input and unnecessary acceleration, we revise the lower bound on exit time to use the maximum value between the earliest feasible exit time computed at t_i^0 (to simplify the notation denoted as t_i^f), and the earliest feasible exit time computed at τ . Thus, the feasible compact set computed at τ is given by

$$\mathcal{T}_i(\tau) = \left[\max \left\{ t_i^f, t_i^{f,\tau} \right\}, \bar{t}_i^{f,\tau} \right]. \quad (12)$$

Problem 1: Each CAV $i \in \mathcal{N}(t)$ at replanning instance τ solves the following optimization problem

$$\begin{aligned} \min_{t_i^f \in \mathcal{T}_i(\tau)} t_i^f \\ \text{subject to:} \\ (4), (7), (8). \end{aligned} \quad (13)$$

To some extent, this replanning provides CAV a feedback mechanism to react to any uncertainties. Ongoing research analyzes the uncertainties and consider these in the planning of CAVs [20].

III. A PRIORITY-AWARE RESEQUENCING

In our previous framework [10], upon entering the control zone, CAV $i \in \mathcal{N}(t)$ solves Problem 1 at $\tau = t_i^0$ by only considering CAVs in the control zone. For the cases in which CAVs enter the control zone simultaneously, the coordinator randomly decides the decision-making order of CAVs. Namely, the order of decision making is based on the order that CAVs entered the control zone, FCFS. We define the *decision sequence* formally as follows.

Definition 2: The sequential decision making of $N(t)$ CAVs is based on the decision sequence that is given by the sequence $s = (s_1, s_2, \dots, s_{N(t)})$ where $s_n \in \mathcal{N}(t)$, $n \in \{1, \dots, N(t)\}$ is the n 'th CAV in the decision making process.

Without a resequencing mechanism, the decision sequence of the $N(t)$ CAVs is given by $s = (1, 2, 3, \dots, N(t))$ which is imposed by the order the CAV enter the control zone, referred to as FCFS sequence. Note that this is different from the order that CAVs cross the intersection, which is determined by the lateral safety constraint (7). Next, we introduce our resequencing framework, which designates the decision sequence at each replanning instance.

Unlike our previous framework, where CAVs only solve their optimization problem upon entering the control zone, in this enhanced framework, CAVs re-solve the optimal control problem at different instances based on new observed information. The observed information of each CAV consists of position and speed of the CAV at the replanning instance, which then can be used as new initial conditions (11) to solve Problem 1. In this section, we introduce a priority-aware resequencing mechanism to find the sequence of decision making based on the minimum exit time from the control zone using scheduling theory.

A scheduling problem is shown by a triplet $(\alpha \mid \beta \mid \gamma)$, where α and β fields describe the machine environment and details of the processing characteristics and constraints, respectively, while γ field describes the objective function. In our problem, the control zone can be considered as a *single machine*, while different CAVs are considered as different jobs. In our problem, we also have *precedence constraint* which requires that a CAV not plan earlier than the physical CAV located in front of it, which we define formally next.

Definition 3: The precedence constraint can be represented by a directed graph $G = (V, E)$, where $V := \mathcal{N}(t)$ is set of all CAVs and $E := \{(i, j) \mid i, j \in V, i \rightarrow j\}$ is the set of all constraints on the order of decision making. Edge $(i, j) \in E$ shows that CAV i should plan earlier than CAV j .

Definition 4: A non-empty subgraph $G_1 = (V_1, E_1)$, where $V_1 \subset V$ and $E_1 \subset E$ is called a **chain** if for each vertex $i \in V_1$, there exist at most a single edge $(i, j) \in E_1, j \in V_1 \setminus \{i\}$.

In a scheduling problem, the *processing time* of a single machine on the job i is denoted by P_i , representing the time that it takes for the machine to process job i . In our case, we consider the processing time of CAV i at replanning instance t to be equal to $\min(\mathcal{T}_i(t))$, i.e., the minimum exit time from the control zone which is independent of the decision sequence. For each job i , a weight $w_i \in \mathbb{R}_{>0}$ describes the importance of job i relative to the other jobs in the system. We consider that the weight of each CAV is inversely proportional to the size of the compact set of its feasible solution. This potentially helps CAVs with smaller feasible space to generate their trajectory first.

Since our goal is to find the optimal decision sequence based on the minimum exit time of the CAVs, we consider the *total weighted completion time* of $N(t)$ CAVs denoted by $J^s = \sum_{i=1}^{N(t)} w_i C_i^s$ as our cost function under decision sequence s , where C_i^s is the sum of processing times of CAV i and other preceding CAVs in the decision sequence s . For example, suppose for two CAV i and j , we have $P_i < P_j$ and $w_i = w_j$. The cost functions for two different decision sequences $s = (i, j)$ and $s' = (j, i)$ are equal to $J^s = w_i \cdot P_i + w_j \cdot (P_i + P_j)$ and $J^{s'} = w_j \cdot P_j + w_i \cdot (P_j + P_i)$, respectively. It is clear that the decision sequence s , which prioritize CAV i over CAV j , has a lower total cost.

Our scheduling problem is denoted by $(1 \mid G \mid \sum_{i=1}^{N(t)} w_i C_i^s)$ which describes a single machine model with precedence constraint G , and the objective is to minimize the total weighted completion time by finding the optimal decision sequence s .

Lemma 1: The precedence constraint's graph of CAVs crossing a single intersection given Assumption 1 consists of multiple disjoint chains.

Proof: From Assumption 1, we do not have any merging or splitting paths, and thus the precedence constraint only exists among CAV i and $j \in \mathcal{N}(t)$ on the same path $\ell \in \mathcal{L}$ such that $\ell_i = \ell_j = \ell$. Thus, among CAVs on path ℓ there exist a chain denoted by $G_\ell \subset G$ such that $\bigcup_{x \in \mathcal{L}} G_x = G$. ■

Definition 5: A ρ -factor of the chain $G_\ell = (V_\ell, E_\ell)$, is denoted by $\rho(G_\ell) \in \mathbb{R}_{>0}$ and for the chain G_ℓ given by

Algorithm 1 Re-Sequencing Algorithm

Input: All available chains $G_\ell = (V_\ell, E_\ell)$, $\ell \in \mathcal{L}$
Output: Decision Sequence $s = (s_1, s_2, s_3, \dots, s_N)$

- 1: **while** $\bigcup_{\ell \in \mathcal{L}} G_\ell$ is not empty **do**
- 2: Update a_ℓ and $\rho(G_\ell)$ for all $G_\ell = (V_\ell, E_\ell)$, $\ell \in \mathcal{L}$
- 3: $\rho_{\max}, \ell^*, a_\ell^* \leftarrow 0$
- 4: **for** $\ell \in \mathcal{L}$ **do**
- 5: **if** $\rho_{\max} < \rho(G_\ell)$ **then**
- 6: \triangleright Find maximum ρ -factor among all chains, the corresponding chain and CAV
- 7: $\rho_{\max} \leftarrow \rho(G_\ell)$; $\ell^* \leftarrow \ell$; $a_\ell^* \leftarrow a_\ell$
- 8: **end if**
- 9: **end for**
- 10: $subSequence \leftarrow \emptyset$
- 11: **while** True **do**
- 12: **if** $\exists i \in V_{\ell^*}$ such that $(i, a_\ell^*) \in E_{\ell^*}$ **then**
- 13: $subSequence.PushFront(i)$ \triangleright Add i to the front of the subsequence
- 14: Remove (i, a_ℓ^*) from E_{ℓ^*}
- 15: Remove i from V_{ℓ^*}
- 16: $a_\ell^* \leftarrow i$
- 17: **else**
- 18: Break
- 19: **end if**
- 20: **end while**
- 21: $sequence.PushBack(subSequence)$ \triangleright Add subsequence to the back of the sequence
- 22: **end while**
- 23: **return** $sequence$

$1 \rightarrow 2 \rightarrow \dots \rightarrow k$ is computed as

$$\rho(G_\ell) = \max_{a \in \{1, \dots, k\}} \left(\frac{\sum_{j=1}^a w_j}{\sum_{j=1}^a P_j} \right) = \frac{\sum_{j=1}^{a^*} w_j}{\sum_{j=1}^{a^*} P_j}, \quad (14)$$

where $a^* \in V_\ell \subset \mathcal{N}(t)$ is called the CAV that determines the ρ -factor of the chain G_ℓ .

The interpretation of the CAV $a^* \in V_\ell$ in the above lemma is that the ratio of weight divided by processing time of the CAV in the chain G_ℓ is increasing from the first CAV in the chain until CAV a^* .

Lemma 2: If CAV $i \in V_\ell \subset \mathcal{N}(t)$ determines the ρ -factor of chain $G_\ell = (V_\ell, E_\ell)$ given by $1 \rightarrow 2 \rightarrow \dots \rightarrow k$, $\ell \in \mathcal{L}$, then there exist an optimal decision sequence that processes CAVs $1, \dots, i$ one after another without any interruption by CAVs from other chains $G_{\ell'}, \ell' \in \mathcal{L} \setminus \{\ell\}$.

Proof: The proof is by contradiction and is similar to that of [21, Lemma 3.1.3.] and follows from [21, Lemma 3.1.2.] by using the results in which it is optimal to process the chain of jobs $1 \rightarrow 2 \rightarrow \dots \rightarrow k$ before the chain of jobs $k+1 \rightarrow \dots \rightarrow n$ if $\frac{\sum_{j=1}^k w_j}{\sum_{j=1}^k P_j} > \frac{\sum_{j=k+1}^n w_j}{\sum_{j=k+1}^n P_j}$. ■

By our resequencing mechanism, at each instance of replanning, CAV i accesses the coordinator and inquires the decision sequence computed using Algorithm 1.

Theorem 1: Under Assumption 1, the decision sequence of $N(t)$ CAVs, which is the optimal solution to $(1 \mid G \mid \sum_{i=1}^{N(t)} w_i C_i^s)$, is computed using Algorithm 1.

Proof: Among all the disjoint chains of all paths, let ρ_{\max} and a_ℓ^* be the maximum ρ -factor and the corresponding CAV determining it, respectively. Namely, ℓ^* is the associated path

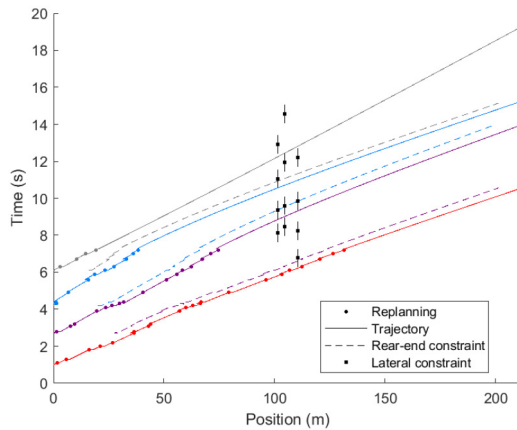


Fig. 2. Time vs position in the presence of deviation.

with ρ_{\max} and a_{ℓ}^* . For every $\ell \in \mathcal{L} \setminus \{\ell^*\}$, we have

$$\rho_{\max} > \frac{\sum_{j \in V_{\ell}}^{a_{\ell}} w_j}{\sum_{j \in V_{\ell}}^{a_{\ell}} P_j}, \quad (15)$$

where a_{ℓ} is the CAV determining the ρ -factor of a chain $G_{\ell} = (V_{\ell}, E_{\ell})$ (lines (3)-(9) in the algorithm). From (15) and [21, Lemma 3.1.2.], the chain G_{ℓ^*} , should be processed first. From Lemma 2, all the CAVs in the chain G_{ℓ^*} , should be processed until CAV a_{ℓ}^* one after another without any interruption by CAVs from other chains (lines (11)-(20) in the algorithm). All processed CAVs get removed from their corresponding chain (lines (14) and (15) in the algorithm), and then the process will be repeated until no CAVs remained unprocessed. ■

IV. SIMULATION RESULTS

We evaluate the effectiveness of our framework in simulation through several scenarios. In all scenarios, we consider CAVs entering the control zone from six different paths shown in Fig. 1, where the length of control zone for straight and turning paths are 212 m and 215 m, respectively. The CAVs enter the control zone with initial speed uniformly distributed between 12 m/s to 17 m/s from each entry with equal traffic volumes varying from 800 to 2400 veh/h. Videos from our simulation analysis can be found at the supplemental site, <https://sites.google.com/view/ud-ids-lab/RPRS>.

For the first scenario, we demonstrate the effects of replanning mechanism to respond to the deviations from the previous planned trajectory. We consider 24 CAVs entering the control zone with the rate of 2,400 veh/h per path, and replanning occurs every time the new CAVs enter the control zone. To only consider the effects of replanning, we set the decision sequence of CAVs at each replanning instance to be based on FCFS sequence. At each replanning instance, we consider CAVs observe their current position and speed with deviation uniformly distributed in the range of $[-2, 2]$ m and $[-0.2, 0.2]$ m/s, respectively. The position trajectory of CAVs traveling from westbound to eastbound is visualized in Fig. 2 in the presence of deviation. The CAVs' positions in this path is denoted by a solid line, while their corresponding rear-end constraints are shown with a dashed line. The CAVs from other paths that have the potential for lateral collision with CAVs

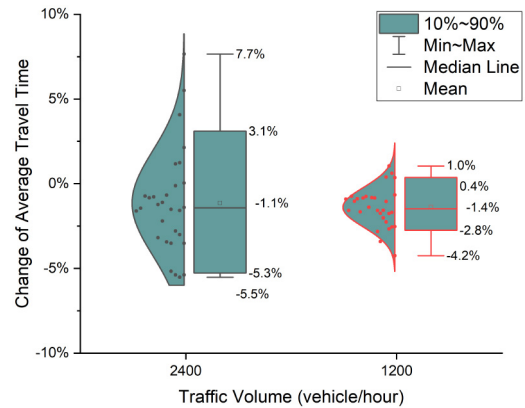


Fig. 3. Change in average travel time compared to the FCFS decision sequence for different traffic volumes.

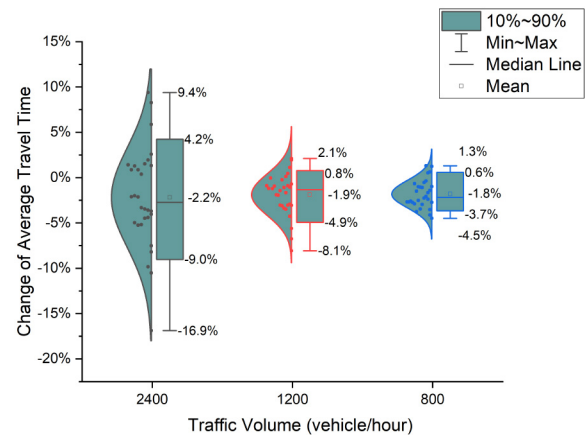


Fig. 4. Change in weighted average travel time compared to the FCFS decision sequence for different traffic volumes.

in this path are shown with a square and vertical bar showing their safety time headway. Figure 2 shows that by using our replanning framework, CAVs respond to the observation made at each replanning point, and they adjust their trajectory to ensure safety.

For the second scenario, we show the change in average travel time of CAVs within our proposed framework compared to the baseline case for different traffic volumes 2,400 veh/h and 1,200 veh/h per path. In the baseline case, CAVs only solve their optimization problem upon entering the control zone based on FCFS, while in our proposed framework, CAVs replan based on the new decision sequence as a new CAV enters the system. In this scenario, we assumed that all CAVs have the same weights, and for each traffic volume, we performed 30 simulations with different random seeds. The results are presented in Fig. 3, and it can be seen how resequencing CAVs affects the average travel time. As the traffic flow increases, the change in average travel time varies more, highlighting the importance of decision sequence in influencing the traffic throughput.

For our last scenario, we demonstrate the change in weighted average travel time of CAVs within our proposed framework compared to the baseline case, at different traffic volumes 2,400 veh/h, 1,200 veh/h, and 800 veh/h in Fig. 4. Similar to the previous scenario, in our proposed framework, CAVs replan based on the new decision sequence as a new

CAV enters the system. We performed 30 simulations with different random seeds for each traffic density. In this scenario, we consider that all CAVs' weights are inversely proportional to the size of their compact set of the feasible solution. After performing 30 different simulations for each traffic flow, our resequencing framework based on the minimum exit time is shown to improve the travel time on average by about 2%.

It should be noted that in 2017, congestion in urban areas across the U.S. led to drivers collectively spending an extra 8.8 billion hours on the road and purchasing an additional 3.3 billion gallons of fuel, ultimately resulting in a \$166 billion expense [22]. Thus, 2% improvement on average travel time in the scale of transportation network by only changing the decision sequence can be quite substantial. Additionally, the main benefit of our approach lies in providing a systematic framework to relax the FCFS sequence in decision making. This would be useful if one needs to prioritize some CAVs to other CAVs, such as giving higher priority to vehicles with higher passenger capacity or emergency vehicles.

By formulating the resequencing problem as a scheduling problem, we find the optimal solution to the scheduling problem. However, this optimal schedule is not the optimal solution, which minimizes the average of actual travel time of all CAVs. The actual travel time of CAVs depends on the decision sequence order, and finding this optimal sequence of decision making is a combinatorial problem, which is an NP-hard problem [15]. However, the algorithm employed in this letter depends on a simple sort which can be done in $O(n \log(n))$ [21]. Thus, if the sole purpose is to improve the average travel time of all CAVs, one can find the decision sequence using our proposed framework and compare it with FCFS policy, and only choose the decision sequence based on the minimum exit time if it improves the performance.

V. CONCLUDING REMARKS AND DISCUSSION

In this letter, we extended the framework we developed earlier [10] for coordination of CAVs at a signal-free intersection to integrate replanning in a time-driven or event-driven manner. This embedded replanning aims at introducing indirect feedback into the coordination framework to respond to the unexpected changes in the system to some extent. Using the theory of the job-shop scheduling problem, we further enhanced our decentralized coordination framework by introducing a priority-aware resequencing mechanism, which designates the order of decision making. The work that we reported on this letter advances the state of the art in a way that relaxes the FCFS decision making sequence of the CAVs. Moreover, in our resequencing framework, we can have different weights representing the priorities for CAVs based on the application. In this letter, we chose the weights of the CAVs to be inversely proportional to the size of their compact set of feasible solutions. We finally demonstrated the effectiveness of our proposed approach through several numerical simulations.

REFERENCES

- [1] 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.
- [2] 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.
- [3] 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.
- [4] H. Xu, Y. Zhang, C. G. Cassandras, L. Li, and S. Feng, "A bi-level cooperative driving strategy allowing lane changes," *Transport. Res. C Emerg. Technol.*, vol. 120, Nov. 2020, Art. no. 102773.
- [5] M. A. Guney and I. A. Raptis, "Scheduling-based optimization for motion coordination of autonomous vehicles at multilane intersections," *J. Robot.*, vol. 2020, 2020, Art. no. 6217409, doi: [10.1155/2020/6217409](https://doi.org/10.1155/2020/6217409). [Online]. Available: <https://www.hindawi.com/journals/jr/2020/6217409/>
- [6] R. Hult, M. Zanon, S. Gros, and P. Falcone, "Optimal coordination of automated vehicles at intersections: Theory and experiments," *IEEE Trans. Control Syst. Technol.*, vol. 27, no. 6, pp. 2510–2525, Nov. 2019.
- [7] B. Chalaki and A. A. Malikopoulos, "Time-optimal coordination for connected and automated vehicles at adjacent intersections," *IEEE Trans. Intell. Transp. Syst.*, early access, Nov. 13, 2021, doi: [10.1109/TITS.2021.3123479](https://doi.org/10.1109/TITS.2021.3123479).
- [8] 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.
- [9] B. Chalaki and A. A. Malikopoulos, "Optimal control of connected and automated vehicles at multiple adjacent intersections," *IEEE Trans. Control Syst. Technol.*, early access, May 31, 2021, doi: [10.1109/TCST.2021.3082306](https://doi.org/10.1109/TCST.2021.3082306).
- [10] A. A. Malikopoulos, L. E. Beaver, and I. V. Chremos, "Optimal time trajectory and coordination for connected and automated vehicles," *Automatica*, vol. 125, no. 6, 2021, Art. no. 109469.
- [11] 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, no. 1, 2019, Art. no. 108563.
- [12] K.-D. Kim and P. R. Kumar, "An MPC-based approach to provable system-wide safety and liveness of autonomous ground traffic," *IEEE Trans. Autom. Control*, vol. 59, no. 12, pp. 3341–3356, Dec. 2014.
- [13] G. R. de Campos, P. Falcone, H. Wymeersch, R. Hult, and J. Sjöberg, "Cooperative receding horizon conflict resolution at traffic intersections," in *Proc. 53rd IEEE Conf. Decis. Control*, 2014, pp. 2932–2937.
- [14] M. Kloock, P. Scheffe, S. Marquardt, J. Maczjowski, B. Alrifaaee, and S. Kowalewski, "Distributed model predictive intersection control of multiple vehicles," in *Proc. IEEE Intell. Transport. Syst. Conf. (ITSC)*, 2019, pp. 1735–1740.
- [15] G. R. de Campos, P. Falcone, and J. Sjöberg, "Autonomous cooperative driving: a velocity-based negotiation approach for intersection crossing," in *Proc. 16th Int. IEEE Conf. Intell. Transport. Syst. (ITSC)*, 2013, pp. 1456–1461.
- [16] B. Alrifaaee, F.-J. Heßeler, and D. Abel, "Coordinated non-cooperative distributed model predictive control for decoupled systems using graphs," *IFAC-PapersOnLine*, vol. 49, no. 22, pp. 216–221, 2016.
- [17] W. Xiao and C. G. Cassandras, "Decentralized optimal merging control for connected and automated vehicles with optimal dynamic resequencing," in *Proc. Amer. Control Conf. (ACC)*, 2020, pp. 4090–4095.
- [18] B. Chalaki, L. E. Beaver, and A. A. Malikopoulos, "Experimental validation of a real-time optimal controller for coordination of cavs in a multi-lane roundabout," in *Proc. 31st IEEE Intell. Veh. Symp. (IV)*, 2020, pp. 504–509.
- [19] A. E. Bryson and Y.-C. Ho, *Applied Optimal Control: Optimization, Estimation and Control*. Boca Raton, FL, USA: CRC Press, 1975. [Online]. Available: <https://doi.org/10.1201/9781315137667>
- [20] B. Chalaki and A. A. Malikopoulos, "Robust learning-based trajectory planning for emerging mobility systems," 2021, *arXiv:2103.03313*.
- [21] M. L. Pinedo, *Scheduling: Theory, Algorithms, and Systems*. Boston, MA, USA: Springer, 2016. [Online]. Available: <https://link.springer.com/book/10.1007%2F978-1-4614-2361-4#about>
- [22] B. Schrank, B. Eisele, and T. Lomax, "2019 Urban Mobility Scorecard," Texas A & M Transport. Inst., INRIX, College Station, TX, USA, Tech. Rep., 2019. [Online]. Available: <https://trid.urb.org/View/1647076>