2020 American Control Conference
Denver, CO, USA, July 1-3, 2020

Decentralized Optimal Merging Control for Connected and Automated Vehicles with Optimal Dynamic Resequencing

Wei Xiao and Christos G. Cassandras

***Abstract*— A complete solution to a decentralized optimal merging problem for Connected and Automated Vehicles (CAVs) was provided in earlier work, based on a First In First Out (FIFO) assumption over a given Control Zone (CZ). In this paper, we relax the FIFO assumption and propose a decentralized Optimal Dynamic Resequencing (ODR) algorithm to further improve the performance of all merging CAVs in terms of time and energy. Specifically, we introduce a Resequencing Zone (RZ) prior to the CZ within which every CAV can execute the ODR algorithm. We determine the latest possible time for triggering ODR so as to minimally affect the CAV's operation. Simulation results show significant ODR benefits in CAV travel times and energy consumption over the merging and main roads, outperforming earlier results under different resequencing schemes.**

I. Introduction

Traffic management at merging points (usually, highway on-ramps) is one of the most challenging problems in transportation systems in terms of safety, congestion, and energy consumption, in addition to being a source of stress for many drivers [1], [2], [3]. The emergence of Connected and Automated Vehicles (CAVs) has the potential to drastically reduce accidents, energy consumption, air pollution, and congestion. An overview of automated intelligent vehicle-highway systems was provided in [4]. For the merging control problem in particular, a number of centralized or decentralized mechanisms has been proposed [5], [6], [7], [8], [2], [9], [10], [11]. In the case of decentralized control, all computation is performed on board each vehicle and shared only with a small number of other vehicles which are affected by it. Optimal control problem formulations are used in some of these approaches, while Model Predictive Control (MPC) techniques are employed to account for additional constraints. The objectives specified for optimal control problems may target the minimization of acceleration as in [9] or the maximization of passenger comfort as in [5], [12]. MPC approaches have been used in [6], [8], as well as in [5] when inequality constraints are added to the originally considered optimal control problem.

In [13], we formulated the merging problem in a decentralized optimal control framework where the objective is to *jointly* minimize the travel time and energy consumption of each CAV subject to a hard speed-dependent safety constraint, as well as speed and acceleration constraints. We derived explicit solutions for the unconstrained case with extensions to optimal trajectories with active constraints and coupled active constraints. From an on-line implementation standpoint, we found that the on-board derivation of such trajectories is very fast (typically, much less than 1sec) in the unconstrained case, but may become more demanding when one or more constraints become active. Thus, when a CAV arrives at the entrance of a Control Zone (CZ) where its optimal trajectory is derived, it is extremely useful to have simple-to-check conditions ensuring that no constraint will become active in the CZ. When these conditions hold, the unconstrained optimal trajectory, derived with minimal computational cost, is guaranteed to also be feasible. A condition of this type was derived in [13], and stronger easy to check conditions for both safety and speed constraints are provided in [14]. Further, to simplify computation while also taking into account the effect of noise in the CAV dynamics, a robust control barrier function method [15] is used in [16], [17].

The work above has been carried out under the assumption that CAVs maintain a First In First Out (FIFO) order upon entering the CZ. This assumption is relaxed in [18] for a 4-way signal-free traffic intersection in which a dynamic resequencing algorithm is implemented whenever a new CAV enters the CZ. This approach has shown significant improvement in travel time when the lengths of CZs from different roads are unbalanced or the traffic flows considerably differ. This comes at the expense of additional energy consumption, as well as driver comfort as it adds jerk (sudden acceleration changes) to those CAVs affected by resequencing; this is because such CAVs are already in the CZ and under optimal control, which resequencing must necessarily disrupt. The coordination of CAVs was formulated as a non-linear programming problem in [19], and thus is time consuming when a lot of CAVs are involved. To address the computational issue, a grouping method was proposed in [20] that introduces a tradeoff between computation and performance.

In this paper, we consider an objective function which combines both travel time and energy consumption as the criterion for resequencing, as opposed to just the travel time in [18]. In addition, we introduce a Resequencing Zone (RZ) before the CZ within which CAVs can collect relevant information from a coordinator located in the merging point. In the RZ, our goal is to determine conditions for triggering an Optimal Dynamic Resequencing (ODR) mechanism for each CAV, where optimality is in the sense of minimally affecting the CAV's operation which we will show implies

*Supported in part by NSF under grants ECCS-1509084, DMS-1664644, CNS-1645681, by AFOSR under grant FA9550-19-1-0158, by ARPA-E's NEXTCAR program under grant DE-AR0000796 and by the MathWorks.

The authors are with the Division of Systems Engineering and Center for Information and Systems Engineering, Boston University, Brookline, MA, 02446, USA {xiaowei,cgc}@bu.edu

triggering ODR as late as possible. The presence of the RZ also allows all CAVs to obtain their optimal controls in the CZ in advance, which has the benefit of causing no additional jerk (i.e., no discontinuity in the controllable acceleration) to CAVs affected by ODR. An additional benefit is that this provides extra time to derive optimal controllers since, as mentioned above, such derivation may no longer be fast when one or more safety and control constraints becomes active. Finally, we develop a decentralized algorithm to implement ODR for each CAV based on the aforementioned trigger conditions in the RZ. The ODR algorithm shows significant improvement when traffic condtions are imbalanced over the main and merging roads.

II. PROBLEM FORMULATION

In this section, we extend the problem formulation in [13] so as to relax the First-In-First-Out (FIFO) rule to the merging problem. The merging problem arises when traffic must be joined from two different roads, usually associated with a main road and a merging road as shown in Fig. 1. We consider the case where all traffic consists of CAVs randomly arriving at the two roads joined at the Merging Point (MP) M where a collision may occur. The segment from the origin O or O' to the merging point M has a length L for both lanes, and is called the Control Zone (CZ). Two connection points C and C' are chosen according to the connection capability of a coordinator for the main and merging lanes, respectively. The segment from the connection point C (or C') to the origin O (or O') has a length L_c for both lanes, and is called the Resequencing Zone (RZ). We assume that CAVs do not overtake each other within the RZ or CZ.

A coordinator is associated with the MP whose function is to maintain RZ and CZ queues of all CAVs regardless of road based on their arrival time at the connection point C (or C') and the origin O (or O') and to enable real-time communication with the CAVs that are in the RZ and CZ, as well as the last one leaving the CZ.

Let $S_c(t)$ be the set of indices for CAVs in the RZ at time t and let $N_c(t)$ be the cardinality of $S_c(t)$. This set is updated in FIFO order when a new CAV arrives at connection point C or C'. In addition, it is updated whenever a CAV is resequenced based on the conditions discussed in the next section. If a CAV arrives at C or C' at time t, it is assigned index $N_c(t)+1$. All CAV indices in $S_c(t)$ decrease by one when the first CAV crosses over to the CZ by passing through the associated origin O or O', and the first CAV is dropped from $S_c(t)$.

Let $S(t)$ be the set of the arrival-time-ordered indices of all CAVs located in the CZ at time t along with the CAV (whose index is 0 as shown in Fig.1) that has just left the CZ. Let $N(t)$ be the cardinality of $S(t)$. Thus, if a CAV arrives at O or O' at time t, all CAV indices in the CZ are resequenced by their arrival time at M. All CAV indices in $S(t)$ decrease by one when a CAV passes over the MP and the vehicle whose index is -1 is dropped.

The vehicle dynamics for each CAV $i \in S(t)$ along the road to which it belongs takes the form

$$\begin{bmatrix} \dot{x}_i(t) \\ \dot{v}_i(t) \end{bmatrix} = \begin{bmatrix} v_i(t) \\ u_i(t) \end{bmatrix}, \tag{1}$$

where $x_i(t)$ denotes the distance to the origin O (O') along the main (merging) road if CAV i is located in the main (merging) road, $v_i(t)$ denotes the velocity, and $u_i(t)$ denotes the control input (acceleration). We consider two objectives for each CAV subject to three constraints, as detailed next.

Objective 1 (Minimize travel time): Let t_i^0 and t_i^m denote the time that CAV $i \in S(t)$ arrives at the origin O or O' and the merging point M, respectively. We wish to minimize the travel time $t_i^m - t_i^0$ for CAV i.

Objective 2 (Minimize energy consumption): We also wish to minimize the energy consumption for each CAV $i \in S(t)$ expressed as

$$J_i(u_i(t)) = \int_{t_i^0}^{t_i^m} \mathcal{C}(u_i(t))dt, \tag{2}$$

where $\mathcal{C}(\cdot)$ is a strictly increasing function of its argument.

Constraint 1 (Safety constraints): Let i_p denote the index of the CAV which physically immediately precedes i in the CZ (if one is present). We require that the distance $z_{i,i_p}(t) := x_{i_p}(t) - x_i(t)$ be constrained by the speed $v_i(t)$ of CAV $i \in S(t)$ so that

$$z_{i,i_p}(t) \geq \varphi v_i(t) + \delta, \quad \forall t \in [t_i^0, t_i^m], \tag{3}$$

where φ denotes the reaction time (as a rule, $\varphi = 1.8$ is used, e.g., [21]). If we define z_{i,i_p} to be the distance from the center of CAV i to the center of CAV i_p, then δ is a constant determined by the length of these two CAVs (generally dependent on i and i_p but taken to be a constant over all CAVs for simplicity).

Constraint 2 (Safe merging): There should be enough safe space at the MP M for a merging CAV to cut in, i.e.,

$$z_{1,0}(t_1^m) \geq \varphi v_1(t_1^m) + \delta. \tag{4}$$

Constraint 3 (Vehicle limitations): Finally, there are constraints on the speed and acceleration for each $i \in S(t)$:

$$\begin{aligned} v_{min} \leq v_i(t) \leq v_{max}, \forall t \in [t_i^0, t_i^m], \\ u_{min} \leq u_i(t) \leq u_{max}, \forall t \in [t_i^0, t_i^m], \end{aligned} \tag{5}$$

where $v_{max} > 0$ and $v_{min} \geq 0$ denote the maximum and minimum speed allowed in the CZ, while $u_{min} < 0$ and $u_{max} > 0$ denote the minimum and maximum control input, respectively.

The common way to minimize energy consumption is by minimizing the control input effort $u_i^2(t)$. By normalizing travel time and $u_i^2(t)$, and using $\alpha \in [0,1]$, we construct a convex combination as follows:

$$\min_{u_i(t)} J_i^1(u_i(t)) = \int_{t_i^0}^{t_i^m} \left(\alpha + \frac{(1-\alpha)\frac{1}{2}u_i^2(t)}{\frac{1}{2}\max\{u_{max}^2, u_{min}^2\}} \right) dt. \tag{6}$$

If $\alpha = 1$, then we solve (6) as a min time problem. Otherwise, by defining $\beta := \frac{\alpha \max\{u_{max}^2, u_{min}^2\}}{2(1-\alpha)}$, we obtain a simplified form:

$$\min_{u_i(t)} J_i^1(u_i(t)) := \beta(t_i^m - t_i^0) + \int_{t_i^0}^{t_i^m} \frac{1}{2}u_i^2(t)dt, \tag{7}$$

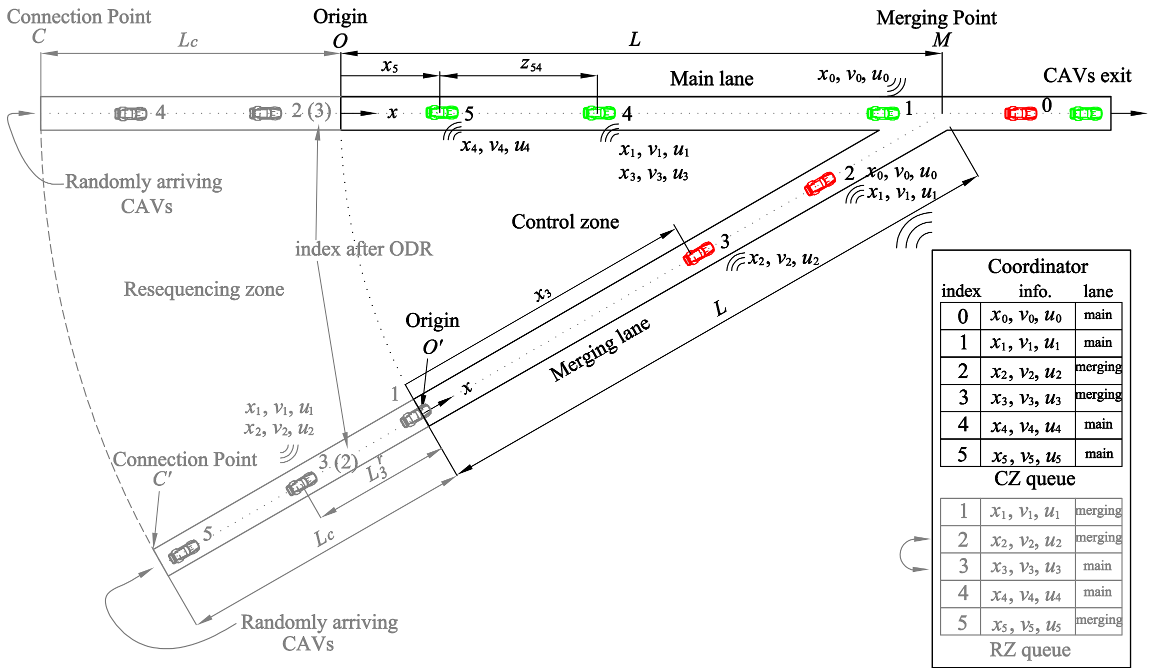

Fig. 1. The merging problem.

where $\beta \geq 0$ denotes a weight factor that can be adjusted to penalize travel time relative to the energy cost.

Our first goal is to determine a control law for each CAV to achieve Objectives 1, 2 subject to Constraints 1-3. Formally, he have:

***Problem** 1:* For each CAV $i \in S(t)$ governed by dynamics (1), determine a control law such that (7) is minimized subject to (1), (3), (4), (5). The initial time t_i^0 and the initial and terminal conditions $x_i(t_i^0) = 0$, $x_i(t_i^m) = L$, and $v_i(t_i^0)$ are given.

Our second goal is to determine an optimal sequence within the RZ queue:

***Problem** 2:* Determine the resequencing trigger conditions for each $i \in S_c(t)$ so as to minimize any effect on the state of i, and subsequently determine the optimal arriving order at the merging point M for all CAVs in the RZ queue.

III. Optimal Dynamic Resequencing

In this section, we first discuss how to solve Problem 2, including a precise way to quantify how we minimize any resequencing effect on the state of CAV i, and then review how to solve Problem 1 following the approach from [13] (so as to make this paper as self-contained as possible).

A. ODR Trigger Conditions

We first address Problem 2 which consists of two parts: first, determine a trigger condition for performing resequencing and then determine the optimal CAV arriving order at the merging point M. Starting with the first part, when a new CAV enters the RZ, it is ready to be resequenced according to certain criteria leading to an Optimal Dynamic Resequencing (ODR) algorithm. However, an optimal sequence is likely to change if the initial speed v_i^0 (at O or O') of a CAV changes. When this happens after the resequencing time, the current optimal sequence and optimal control profiles of all CAVs (which are already evaluated prior to the resequencing time) affected in the RZ will no longer be optimal. This motivates us to impose the following constraint:

Assumption 1: Let t_i^r be the time instant when CAV $i \in S_c(t)$ performs ODR. Then, its speed is constant until it reaches the CZ, i.e., $v_i(t) = v_i(t_i^r)$ $\forall t \in [t_i^r, t_i^0]$.

Assumption 1 provides the motivation for seeking to delay resequencing in the RZ as long as possible so as not to affect a CAV's speed prior to the CZ. Thus, we formulate an optimization problem which defines the triggering condition for invoking ODR. To formulate this problem, let $S_i(t) \subseteq S_c(t)$ denote the index set of all possible CAVs that could be surpassed by i and let t_i^c denote the time when i arrives at the connection point C so as to enter the RZ. Recall that i_p is the index of the CAV that immediately precedes CAV i if such a CAV exists; if such a CAV does not exist, we set $i_p = 0$, otherwise clearly $i_p < i$. Since i cannot surpass i_p (if it exists), $S_i(t)$ is given by

$$S_i(t) := \{i_p + 1, \ldots, i - 1\}. \tag{8}$$

Note that every $j \in S_i(t)$ has already performed ODR, therefore its speed is constant and any other information related to it (such as its position $x_j(t)$ within its road in the RZ) is also known. Therefore, the optimal control problem solution described in Section III-C may now be obtained as soon as a CAV i is resequenced at t_i^r rather than waiting until it reaches O or O' at $t_i^0 > t_i^r$.

We now seek to minimize the position $-x_i(t_i^r)$ for each CAV i when it performs ODR, i.e., we formulate the first part of Problem 2 as follows:

$$\min_{t_i^r \in [t_i^c, t_i^0]} -x_i(t_i^r) \equiv J_i^2(t_i^r), \tag{9}$$

subject to

$$x_j(t_i^r) \leq 0, \forall j \in S_i(t),$$

where $x_j(t)$ denotes the position of CAV $j \in S_i(t)$ along its road and is known to the coordinator.

Recall that $t^0_{i_p+1}$ (known by Assumption 1) denotes the time instant when the $(i_p + 1)$th CAV (the first when $i_p = 0$) in the RZ arrives at the origin O or O'. The following theorem offers the optimal solution of the objective (9) while guaranteeing that an optimal sequence for all CAVs in the CZ can be determined.

Theorem 1: The optimal solution for ODR trigger condition is given by

$$t_i^{r*} = t^0_{i_p+1}, \tag{10}$$

such that the optimal sequence is guaranteed to be found.

Proof: Since we have assumed that all CAVs in the CZ and RZ cannot overtake each other, then CAV i cannot overtake i_p. In terms of CAV i_p, there are two cases: (i) CAV i does not have an i_p in the RZ (i.e., $i_p = 0$), in which case all CAVs in the RZ are in the road which does not include i, hence CAV i may overtake all of them. Thus, CAV i minimizes $-x_i(t_i^r)$ when the first vehicle in the RZ arrives at origin O or O' such that the optimal sequence is guaranteed to be found while also satisfying Assumption 1. (ii) CAV i does have an i_p in the RZ (i.e., $i_p > 0$). In this case, CAV i can only overtake CAV j such that $i < j < i_p$, and $-x_i(t_i^r)$ is minimized when CAV $i_p + 1$ arrives at the origin O or O'.

We have thus shown that the ODR trigger condition is given by $t = t_i^{r*} = t^0_{i_p+1}$. ■

B. Decentralized ODR

Let $L_i^r := -x_i(t_i^r)$, $i \in S_c(t)$. Under Assumption 1, we know the arrival times of all ODR-affected CAVs and their inital speeds at the origin O or O', i.e.,

$$t_j^0 = t_j^r + L_j^r / v_j^0, \qquad v_j^0 = v_j(t_j^r), \quad j \in S_i(t)$$

are known constants. When an ODR command is triggered for CAV $i \in S_c(t)$, i.e. $t = t_i^{r*}$, the coordinator will inform all CAVs that are affected by ODR in the RZ to re-evaluate their optimal trajectories in the CZ according to a given updated sequence and return their optimal objective function values to the coordinator; the coordinator will then send all these optimal objective function values to CAV i which determines the optimal sequence.

Let N_i denote the cardinality of S_i (where we omit time arguments for simplicity) and observe that

$$N_i = i - i_p - 1. \tag{11}$$

Then, we want to find the optimal number k, $0 \leq k \leq N_i$, of CAVs that i wishes to surpass in order to obtain the optimal sequence. Since every ODR-affected CAV depends on k, then the objective (7) depends on k if we do ODR, i.e., we have that $J_j^1(k, t_j^0, v_j^0)$, for $j \in S_i(t)$ (i.e., Problem 1) is also a function of k. Therefore, the second part of Problem 2 can be reformulated as a minimization of the overall objective function $J_i(k)$ given by

$$\min_k J_i(k) = \sum_{j=i-N_i}^{i-1} J_j^1(k, t_j^0, v_j^0). \tag{12}$$

subject to $0 \leq k \leq N_i$.

The following lemma establishes the convexity of $J_i(k)$ which will subsequently facilitate the solution of this problem.

Lemma 1: If every CAV $j \in S_i(t)$ has a unique optimal control solution in the CZ, then $J_i(k)$ is a convex function of k.

Proof: Since the optimal solution in the CZ for CAV $j \in S_i(t)$ is obtained by taking derivatives of the Hamiltonian function and every CAV has a unique optimal control solution, it follows that $J_j^1(k, t_j^0, v_j^0)$ is a convex function of the number k of CAVs to be surpassed. Since the summation of convex functions is also a convex function, it follows that $J_i(k)$ is a convex function of k. ■

Once we know that $J_i(k)$ is a convex function of k, there is no need to consider all possible k (as done in [18] for an intersection problem). Instead, we can use the main idea of fast sorting (the best among sort algorithms) to find the optimal k. Specifically, we set $k = N_i/2$ in the first iteration, and then determine $J_i(k)$, $J_i(k-1)$, $J_i(k+1)$. If $J_i(k-1) < J_i(k) < J_i(k+1)$, then we set $k = N_i/4$, and if $J_i(k-1) > J_i(k) > J_i(k+1)$, we set $k = 3N_i/4$, otherwise, $k = N_i/2$ is the optimal solution. We continue this process until we determine the optimal $J_i^*(k^*)$. The resulting ODR procedure is shown in Algorithm 1.

The complexity of Algorithm 1 is $O(N_i)$, i.e., a CAV $j \in S_i(t)$ may need to have $O(N_i)$ processes for determining an optimal solution for each ODR trigger.

C. Optimal Control in CZ

Recall that i_p is the index of the CAV that immediately (in the same road) precedes CAV $i \in S(t)$. We need to distinguish between the following two cases:

- (i) $i_p = i-1$, *i.e.,* i_p is the CAV immediately preceding i in the CZ queue (such as CAVs 3 and 5 in Fig. 1), and
- (ii) $i_p < i-1$ (such as CAVs 2 and 4 in Fig.1), which implies CAV $i-1$ is in a different lane from i.

We can solve Problem 1 for all $i \in S(t)$ in a decentralized way, in the sense that CAV i can solve it using only its own local information (position, velocity and acceleration) along with that of its "neighbor" CAVs $i-1$ and i_p (in case (ii) only). Observe that if $i_p = i-1$, then (4) is a redundant constraint. Otherwise, we need to consider (3) and (4) independently.

Assuming that (3) and (5) remain inactive over $[t_i^0, t_i^m]$, we can obtain the unconstrained optimal solution as described next.

Algorithm 1: Fast ODR algorithm

Input: L_i^{r*} or t_i^{r*}, $S_c(t_i^{r*})$
Output: $S_c^*(t_i^{r*})$
$k^* = -1$;
ODR-affected CAV number $N_i = i - i_p - 1$;
k search space is in $S_0 := \{0, 1, \ldots, N_i\}$;
CAV surpassing number $k \leftarrow N_i/2$;
S_0 is partitioned into $S_1 := \{0, 1, \ldots, k-1\}$ and $S_2 := \{k+1, \ldots, N_i\}$;
while $k \neq k^*$ **do**
  $N_i \leftarrow N_i/2$;
  Find $J_i(k), J_i(k-1), J_i(k+1)$;
  if $J_i(k-1) < J_i(k) < J_i(k+1)$ **then**
    $S_0 \leftarrow S_1$;
    S_0 is equally partitioned into S_1 and S_2, $k \leftarrow N_i/2$;
  else
    if $J_i(k-1) > J_i(k) > J_i(k+1)$ **then**
      $S_0 \leftarrow S_2$;
      S_0 is equally partitioned into S_1 and S_2, $k \leftarrow k + N_i/2$;
    else
      $k^* = k$
    end
  end
  if $k = 0$ *or* $k = N_i$ **then**
    $k^* = k$;
  end
end
Resequence $S_c(t_i^{r*})$ according to k^* and get $S_c^*(t_i^{r*})$;

1) Unconstrained optimal control: Since the constraints (3) and (5) are assumed to be inactive in the CZ, it follows from the analysis in [13] that the unconstrained optimal solution is

$$\begin{aligned} u_i^*(t) &= a_i t + b_i, \\ v_i^*(t) &= \frac{1}{2} a_i t^2 + b_i t + c_i, \\ x_i^*(t) &= \frac{1}{6} a_i t^3 + \frac{1}{2} b_i t^2 + c_i t + d_i \end{aligned} \tag{13}$$

where a_i, b_i, c_i and d_i are integration constants. The four constants a_i, b_i, c_i, d_i and the optimal terminal time t_i^m in (7) in the case $i-1 = i_p$ are obtained by solving the following five nonlinear algebraic equations:

$$\begin{aligned} &\frac{1}{2} a_i \cdot (t_i^0)^2 + b_i t_i^0 + c_i = v_i^0, \\ &\frac{1}{6} a_i \cdot (t_i^0)^3 + \frac{1}{2} b_i \cdot (t_i^0)^2 + c_i t_i^0 + d_i = 0, \\ &\frac{1}{6} a_i \cdot (t_i^m)^3 + \frac{1}{2} b_i \cdot (t_i^m)^2 + c_i t_i^m + d_i = L, \\ &a_i t_i^m + b_i = 0, \\ &\beta + \frac{1}{2} a_i^2 \cdot (t_i^m)^2 + a_i b_i t_i^m + a_i c_i = 0. \end{aligned} \tag{14}$$

In the case $i-1 > i_p$, CAV i_p, which physically precedes $i \in S(t)$, is different from $i-1$ and is, therefore, in a different lane from i. This implies that we need to consider the safe merging constraint (4) at $t = t_i^m$. Following the analysis in [13], we obtain the same optimal solution as in (13), where a_i, b_i, c_i, d_i and t_i^m are now determined by another five nonlinear algebraic equations (more complicated than (14), and are skipped).

Since we aim for the solution to Problem 1 to be obtained on-board each CAV, it is essential that the computational cost of solving (14) (or another five nonlinear algebraic equations in the case $i-1 > i_p$) be minimal. If MATLAB is used, it takes less than 1 second to get the solution (Intel(R) Core(TM) i7-8700 CPU @ 3.2GHz×2).

2) Constrained optimal control: When the constraints (3), (4), (5) become active, a complete OC solution can still be obtained [13], [22], but the computation time varies between 3 and 30 seconds depending on whether i_p is also safety-constrained or not.

IV. Implementation and Case Study

We have implemented the proposed ODR algorithm in MATLAB. The CAVs randomly arrive following Poisson processes at both roads. We studied two different cases: equal arrival rates at both roads; a 3:1 arrival rate ratio for the merging road relative to the main road. In addition, we studied different optimal trade-off parameter α values in (6). The model parameters used in the simulated system are $\varphi = 1.8s, \delta = 9m, L = 400m, L_c = 200m, v_{max} = 30m/s, v_{min} = 0m/s, u_{max} = -u_{min} = 3.924m/s^2$.

We compared the results from this paper, with those of the dynamic resequencing (DR) approach in [18], as well as the results based on a FIFO queue from [13]. The comparison results are shown in Tab. I.

As expected, the ODR method has the best average objective function value in all cases among these three approaches, as shown in Tab. I. The times that CAVs do resequencing (i.e., $k^* \neq 0$ in Algo. 1) are also improved compared with the DR approach [18] in all cases. In order to also demonstrate the dramatic improvement in jerk (hence, improved comfort) for CAVs which are affected by dynamic reseqencing, the control profiles of an example CAV under the ODR approach in this paper and the DR approach in [18] are shown in Fig. 2. As seen in this figure, it is obvious that the control of this CAV affected by resequencing experiences a jump under the DR approach, thus causing significant jerk to it.

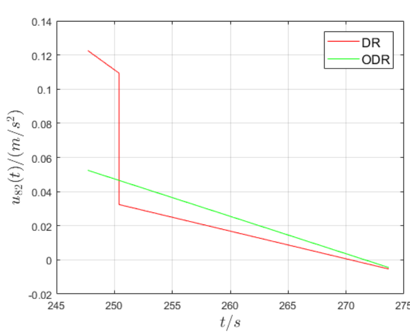

Fig. 2. Control profile comparisons between DR [18] and ODR for one example CAV in the 3:1 arrival ratio case under trade-off $\alpha = 0.01$.

TABLE I

COMPARISONS BETWEEN THE RESULTS FROM THIS PAPER, THE RESULTS WITH DR FROM [18] AND THE RESULTS WITH FIFO FROM [13]

Method	Arrival	α	Avg. obj.	Avg. time	Avg. energy	Reseq./total	Merg.: Reseq./ total
FIFO	Equal arrival	0.25	36.4909	17.0472	**4.9069**	0/91	0/45
DR			36.3887	16.9230	5.0893	5/91	5/45
ODR			**36.3592**	**16.9190**	5.0603	**9**/91	**6**/45
FIFO	3:1 ratio		36.7389	17.0559	**5.2152**	0/90	0/27
DR			36.6471	**16.8652**	5.5822	4/90	3/27
ODR			**36.6135**	16.8663	5.5346	**5**/90	**4**/27
FIFO		0.01	2.0631	25.4881	0.1015	0/90	0/27
DR			1.9699	24.9176	0.0518	12/90	9/27
ODR			**1.9626**	**24.8991**	**0.0458**	**15**/90	**10**/27
FIFO	3:1 ratio, merg. $4m/s$ higher in avg. v_i^0		2.2915	24.8383	0.3828	0/90	0/27
DR			1.9176	**23.9250**	0.0761	17/90	17/27
ODR			**1.9106**	24.0227	**0.0615**	**18**/90	**18**/27

The computation time in MATLAB (Intel(R) Core(TM) i7-8700 CPU @ 3.2GHz$\times$2) under the FIFO approach for each CAV is around 1 second (unconstrained optimal), and is the smallest one among these three approaches. The computation time when using the ODR is better than DR due to Lemma 1 and the use of the fast search algorithm in Algo. 1.

Finally, the ODR approach also has the advantage of computing an optimal solution to the merging problem for each CAV in advance, i.e., before the CAV enters the CZ. This is because the CAV's arrival time and speed upon arriving at the origin O or O' are now already known at time $t_i^r < t_i^0$. This is important when the solution involves one or more active constraints which require a longer computation time than the unconstrained case.

V. CONCLUSIONS

We have proposed an optimal dynamic resequencing framework in order to relax the FIFO assumption made in earlier work which may diminish the performance attainable by all CAVs in a traffic merging problem. This is especially true when the traffic conditions are unbalanced over the main and merging road. The introduction of a RZ allows all CAVs to obtain their associated optimal merging controllers before they enter the CZ, minimizes the effect of resequencing on the state of a CAV (hence, its normal operation), and it improves performance for all CAVs compared with FIFO-based operation or earlier dynamic resequencing approaches. Future work will focus on multi-road merging problems that can further improve CAV performance.

REFERENCES

[1] B. Schrank, B. Eisele, T. Lomax, and J. Bak. (2015) The 2015 urban mobility scorecard. Texas A&M Transportation Institute. [Online]. Available: http://mobility.tamu.edu

[2] M. Tideman, M. van der Voort, B. van Arem, and F. Tillema, "A review of lateral driver support systems," in *Proc. IEEE Intelligent Transportation Systems Conference*, pp. 992–999, Seatle, 2007.

[3] D. D. Waard, C. Dijksterhuis, and K. A. Broohuis, "Merging into heavy motorway traffic by young and elderly drivers," *Accident Analysis and Prevention*, vol. 41, no. 3, pp. 588–597, 2009.

[4] P. Varaiya, "Smart cars on smart roads: problems of control," *IEEE Transactions on Automatic Control*, vol. 38, no. 2, pp. 195–207, 1993.

[5] I. A. Ntousakis, I. K. Nikolos, and M. Papageorgiou, "Optimal vehicle trajectory planning in the context of cooperative merging on highways," *Transportation Research, Part C*, pp. 464–488, 2016.

[6] W. Cao, M. Mukai, and T. Kawabe, "Cooperative vehicle path generation during merging using model predictive control with real-time optimization," *Control Engineering Practice*, vol. 34, pp. 98–105, 2015.

[7] V. Milanes, J. Godoy, J. Villagra, and J. Perez, "Automated on-ramp merging system for congested traffic situations," *IEEE Trans. on Intelligent Transport. Systems*, vol. 12, no. 2, pp. 500–508, 2012.

[8] M. Mukai, H. Natori, and M. Fujita, "Model predictive control with a mixed integer programming for merging path generation on motor way," in *Proc. IEEE Conf. on Control Tech. and Applications*, 2017.

[9] J. Rios-Torres, A. Malikopoulos, and P. Pisu, "Online optimal control of connected vehicles for efficient traffic flow at merging roads," in *Proc. IEEE 18th International Conference on Intelligent Transportation Systems*, pp. 2432–2437, 2015.

[10] G. Raravi, V. Shingde, K. Ramamritham, and J. Bharadia, *Merge algorithms for intelligent vehicles. In: Sampath, P., Ramesh, S. (Eds.), Next Generation Design and Verification Methodologies for Distributed Embedded Control Systems*. Waltham, MA: Springer, 2007.

[11] R. Scarinci and B. Heydecker, "Control concepts for facilitating motorway on-ramp merging using intelligent vehicles," *Transport Reviews*, vol. 34, no. 6, pp. 775–797, 2014.

[12] C. Rathgeber, F. Winkler, X. Kang, and S. Muller, "Optimal trajectories for highly automated driving," *Intl. Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Eng.*, vol. 9, no. 6, pp. 946–952, 2015.

[13] W. Xiao and C. G. Cassandras, "Decentralized optimal merging control for connected and automated vehicles," in *Proc. of the American Control Conference*, pp. 3315–3320, 2019.

[14] ——, "Conditions for improving the computational efficiency of decentralized optimal merging controllers for connected and automated vehicles," in *Proc. of 58th IEEE Conference on Decision and Control*, pp. 3158–3163, 2019.

[15] W. Xiao and C. Belta, "Control barrier functions for systems with high relative degree," in *Proc. of 58th IEEE Conference on Decision and Control*, pp. 474–479, 2019.

[16] W. Xiao, C. Belta, and C. G. Cassandras, "Decentralized merging control in traffic networks: A control barrier function approach," in *Proc. ACM/IEEE International Conference on Cyber-Physical Systems*, pp. 270–279, 2019.

[17] 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 22nd Intelligent Transportation Systems Conference*, pp. 3162–3167, 2019.

[18] Y. J. Zhang and C. G. Cassandras, "A decentralized optimal control framework for connected automated vehicles at urban intersections with dynamic resequencing," in *Proc. 57th IEEE Conference on Decision and Control*, pp. 217–222, 2018.

[19] R. Hult, M. Zanon, S. Gros, and P. Falcone, "Optimal coordination of automated vehicles at intersections with turns," in *2019 18th European Control Conference*, pp. 225–230, 2019.

[20] H. Xu, S. Feng, Y. Zhang, and L. Li, "A grouping-based cooperative driving strategy for cavs merging problems," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 6, pp. 6125–6136, 2019.

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

[22] Y. J. 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. 1–16, 2019.