2018 European Control Conference (ECC)
June 12-15, 2018. Limassol, Cyprus

Distributed control algorithm for vehicle coordination at traffic intersections

Jiahe Shi[1], Yi Zheng[1], Yuning Jiang[1], Mario Zanon[2], Robert Hult[2], Boris Houska[1]

***Abstract*— This paper proposes a distributed closed-loop control algorithm for optimal coordination of autonomous vehicles at traffic intersections. The main contribution of the paper is a distributed plug-and-play closed-loop optimal control scheme with rear-end collision avoidance constraints enforced on each lane, which maintains recursive feasibility under the assumption that communication between neighboring vehicles is possible. The method is closely related to model predictive control, but at each sampling time new vehicles are allowed to enter the modeled region around the intersection while other vehicles are leaving. In contrast to human drivers, autonomous vehicles can collaboratively form the deceleration strategy before the intersection. Our numerical results indicate that, under certain assumptions, it is optimal for vehicles not allowed to directly pass the intersection to slow down much before the intersection area and then accelerate at the right time so that they can travel through the intersection faster.**

I. INTRODUCTION

The development of autonomous vehicles opens the door toward completely new approaches for infrastructure management and traffic-flow control in cities. In fact, autonomous vehicles have an enormuous potential to reduce traffic congestions, accidents, and fatalities [4]. Because modern vehicles can communicate via wireless networks, many researchers have proposed methods for coordinating automuous vehicles at intersections [6], [7], [18] as well as methods for reliable communication between vehicles [7], [8], [17]. Moreover, in [1], [2], [5] methods for verifying safety have been proposed by using a supervised network. Similarly, in [10] methods from the field of hybrid system theory have been used to address the safe traffic control problem.

Traffic control problems at an intersection can be formulated as structured optimal control problems. Here, the dynamics of the vehicles are often modeled by simple double integrators but the security and collision avoidance constraints typically lead to a non-convex optimization problem, which is challenging to solve—even if the order of the vehicles passing the intersection is pre-determined [12]. Moreover, one is interested in solving this optimal control problem in a distributed way using onboard computing devices on each of the vehicles while avoiding too much communication overhead between the vehicles. There exist several articles, which have addressed this problem. For example, in [13] a hierarchical decomposition of the original problem based on sequential quadratic programming has been proposed. The corresponding method has been generalized in [21] using asynchronous control and communication schemes. Moreover, in [14] an alternating direction augmented Lagrangian based inexact Newton method (ALADIN) [11], [16] has been used to solve the traffic optimal control problem in a distributed way by negotiating a time schedule between the vehicles. Other authors [15], [19] have developed convex relaxations of the optimal traffic control problem such that existing convex decomposition methods can be applied, although, in general, this leads to sub-optimal solutions.

Traffic optimal control problems can also be solved online whenever new measurements are available, which leads to a model predictive control (MPC) strategy for traffic control [6], [3]. An example for a fully parallelizable MPC approach for traffic control MPC can be found in [15]. As it will also be discussed further in this paper, the corresponding optimal control based closed-loop schemes are, however, different from the standard MPC controllers in the sense that the problem structure changes online. For example, if new cars arrive in the intersection area, the number of total differential states in the optimal control network increases—a structural change that is not covered by standard model predictive control theory [20]. Similarly, vehicles unplug from the network after leaving the surrounding of the intersection.

The contribution of this paper is twofold. Firstly, while existing traffic optimal control formulations [12], [14], [21] already take collision avoidance constraints on the intersection into account, this paper focusses on optimal control problem formulations that additionally model rear-end collision constraints of neighboring vehicles in each lane. This is nontrivial, because these rear-end collision constraints need to be enforced at all times and lead to additional coupling terms between neighbors in each lane. In particular, Section II focusses on the exact reformulation of semi-infinite rear-end collision constraints as well as the structural properties of the associated coupled optimal control problems. And secondly, Section III-B proposes a recursively feasible closed-loop control and communication scheme, which is based on the distributed optimization algorithm ALADIN [11] and which can ensure recursive feasibility of the traffic control problem even if newly arriving cars are plugged into the network while other vehicles are leaving the intersection area. Section IV illustrates the practical performance of this control scheme for various scenarios. In particular, we discuss why

[1]School of Information Science and Technology, ShanghaiTech University, Shanghai, China. shijh, zhengyi1, jiangyn, borish@shanghaitech.edu.cn

[2]Department of Signals and Systems, Chalmers University of Technology, Göteborg, Sweden mario.zanon, robert.hult@chalmers.se

978-3-9524-2699-9 ©2018 EUCA

it is important to model rear-end collision constraints during queueing phases by analyzing a 3-lane scenario with 5 cars, where 2 vehicles need to yield to others. Section V concludes the paper.

II. A DYNAMIC MODEL FOR VEHICLES AT INTERSECTIONS

This section proposes a multi-lane model for autonomous vehicle control at traffic intersections, as visualized in Figure 1 for a scenario with four lanes.

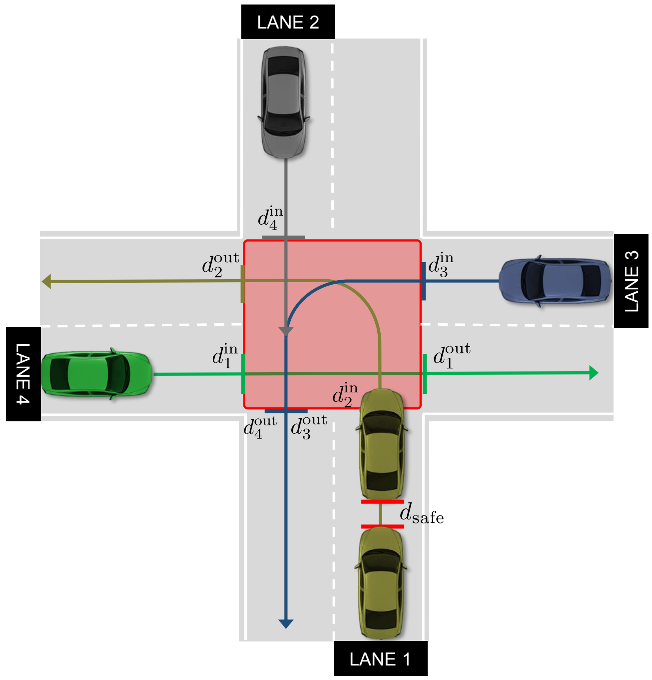

Fig. 1: Sktech of an intersection with four entrance lanes and five cars.

Similar to [13], this algorithm takes the pre-determined orders of vehicles entering the intersection as an input.

A. Control discretization

We use the index $i \in \{1, \ldots, M\}$ to enumerate all M vehicles that are currently inside the intersection area or in front of it. Let t_i^{in} and t_i^{out} denote the times at which the i-th vehicle enters and leaves the intersection. We divide the interval $[0, t_i^{\text{in}}]$ into K_i sub-intervals, where

$$0 = K_0 < K_1 < K_2 < \ldots < K_M \ .$$

Similarly, the interval $[t_i^{\text{in}}, t_i^{\text{out}}]$ is divided into L_i sub-intervals, i.e., such that the i-th vehicle has $N_i = K_i + L_i$ control intervals in total. Let us introduce the shorthands

$$T_i = [t_1^{\text{in}},\, t_2^{\text{in}},\, t_3^{\text{in}},\, \ldots,\, t_i^{\text{in}},\, t_i^{\text{out}}]^{\mathsf{T}} \in \mathbb{R}^{i+1}$$

and

$$h_{i,k}(T_i) = \begin{cases} \frac{t_1^{\text{in}} - t_0^{\text{in}}}{K_1 - K_0} & \text{if } k \in \mathbb{Z}_{K_0}^{K_1 - 1} \ , \\ \vdots & \vdots \\ \frac{t_i^{\text{in}} - t_{i-1}^{\text{in}}}{K_i - K_{i-1}} & \text{if } k \in \mathbb{Z}_{K_{i-1}}^{K_i - 1} \ , \\ \frac{t_i^{\text{out}} - t_i^{\text{in}}}{L_i} & \text{otherwise} \end{cases}$$

for all $i \in \{1, \ldots, M\}$, all $k \in \{0, \ldots, N_i - 1\}$, where we set $t_0^{\text{in}} = 0$ and $\mathbb{Z}_a^b$ denotes $[a, b] \cap \mathbb{Z}$. The time grid is globally syncronized. Next, we introduce a grid point function

$$t_{i,k}(T_i) = \sum_{j=0}^{k-1} h_{i,j}(T_i) \ ,$$

which depends on the parameter vector T_i. Clearly, if the acceleration, $u_i(t)$, of the i-th vehicle is piecewise constant with respect to these grid points, i.e.

$$u_i(t) = \begin{cases} a_{i,1} & \text{if } t \in [t_{i,1}(T_i), t_{i,2}(T_i)] \ , \\ \vdots & \vdots \\ a_{i,N_i} & \text{if } t \in [t_{i,N_i}(T_i), t_{i,N_{i+1}}(T_i)] \ , \end{cases} \tag{1}$$

where $a_{i,j}$ denotes the value of piecewise constant acceleration of car i on interval j. Its associated velocity and position (on a pre-determined path) at the grid points are given by the discrete-time recursion

$$\begin{aligned} &\forall k \in \{0, \ldots, N_i - 1\}, \\ &v_{i,k+1}(a_i, T_i) \;=\; v_{i,k}(a_i, T_i) + a_{i,k} h_{i,k}(T_i) \ , \\ &p_{i,k+1}(a_i, T_i) \;=\; p_{i,k}(a_i, T_i) + v_{i,k}(a_i, T_i) h_{i,k}(T_i) \\ &\qquad\qquad + \tfrac{1}{2} a_{i,k} h_{i,k}(T_i)^2 \ . \end{aligned}$$

Here, we assume that the initial positions and velocities,

$$p_{i,0}(a_i, T_i) = \hat{p}_i \quad \text{and} \quad v_{i,0}(a_i, T_i) = \hat{v}_i \ ,$$

are given. Notice that $v_{i,k}$ is a jointly affine function with respect to a_i and T_i. Similarly, the function $p_{i,k}$ is a quadratic form in (a_i, T_i). The associated continuous-time function for the position and velocity of the i-the vehicle can be recovered as

$$\begin{aligned} &\forall\, t \in [t_{i,k}(T_i), t_{i,k+1}(T_i)], \\ &\begin{cases} V_i(t, a_i, T_i) \;=\; v_{i,k}(a_i, T_i) + a_{i,k}(t - t_{i,k}(T_i)) \ , \\ P_i(t, a_i, T_i) \;=\; p_{i,k}(a_i, T_i) \\ \qquad + v_{i,k}(a_i, T_i)(t - t_{i,k}(T_i)) \\ \qquad + \tfrac{1}{2} a_{i,k}(t - t_{i,k}(T_i))^2 \ . \end{cases} \end{aligned} \tag{2}$$

The function $V_i(\cdot, a_i, T_i)$ is piecewise linear in time while the position $P_i(\cdot, a_i, T_i)$ is a piecewise quadratic function in time.

B. Collision avoidance

As originally suggested in [14], one can enforce that no collisions occur inside the intersection area by introducing the collision avoidance constraints

$$p_{i,K_i}(a_i, T_i) = d_i^{\text{in}} \ , \ p_{i,N_i}(a_i, T_i) = d_i^{\text{out}} \ , \ t_i^{\text{out}} \leq t_{i+1}^{\text{in}} \ , \tag{3}$$

such that there is at any given time t at most one vehicle inside the intersection area. Here, d_i^{in} and d_i^{out} denote the start and the end of the intersection area on the pre-determined path of the i-th vehicle. In contrast to the models in [13], [14], the focus of this paper is, however, on ensuring that no rear-end collisions occur.

Let $\mathbb{M}$ be the set of ordered pairs of indices of neighbors in the same lane such that, if vehicle i and vehicle j, with

$i < j$, are neighbors in the same lane, then $(i,j) \in \mathbb{M}$. Now, there are no rear-end collisions possible, if the semi-infinite constraint

$$\begin{aligned} &\forall (i,j) \in \mathbb{M},\ \forall t \in [0, t_i^{\mathrm{in}}], \\ &P_i(t, a_i, T_i) - P_j(t, a_j, T_j) > d_{\mathrm{safe}}\ , \end{aligned} \tag{4}$$

is satisfied. Here, $d_{\mathrm{safe}} \geq 0$ is a given minimum security distance between the centers of two vehicles. Because this semi-infinite constraint is in this form computationally intractable, we need to introduce the following equivalent form,

Lemma 1 *The semi-infinite inequality*

$$\forall \tau \in [-1,1], \quad \alpha\tau^2 + \beta\tau + \gamma > 0 \tag{5}$$

is satisfied if and only if there exists a $\delta > 0$ *with*

$$\delta + \frac{1}{4}\frac{\beta^2}{\alpha+\delta} < \gamma \ \text{ and } \ \alpha + \delta > 0\ . \tag{6}$$

Proof. The semi-infinite inequality (5) is equivalent to

$$0 < \min_{\tau^2 \leq 1} \quad \alpha\tau^2 + \beta\tau + \gamma\ .$$

Thus, the tight version of the S-procedure for non-convex quadratically constrained quadratic programs [9] can be applied to find the equivalent dual condition

$$0 < \sup_{\delta > 0} \quad \gamma - \delta - \frac{1}{4}\frac{\beta^2}{\alpha + \delta} \quad \text{s.t. } 0 < \alpha + \delta\ .$$

This inequality, in turn, is equivalent to the statement of the lemma. □

In the following, we introduce a small abuse of notation by writing $h_{i,k}$ instead of $h_{i,k}(T_i)$, $v_{i,k}$ instead of $v_{i,k}(a_i,T_i)$, and $p_{i,k}$ instead of $p_{i,k}(a_i,T_i)$, i.e., not all dependencies are shown. Next, we use the shorthands

$$\begin{aligned} \alpha_{i,j,k} &= \frac{h_{i,k}^2}{8}(a_{i,k} - a_{j,k}), \\ \beta_{i,j,k} &= \frac{h_{i,k}^2}{4}(a_{i,k} - a_{j,k}) + \frac{h_{i,k}}{2}(v_{i,k} - v_{j,k}), \\ \gamma_{i,j,k} &= \frac{h_{i,k}^2}{8}(a_{i,k} - a_{j,k}) + \frac{h_{i,k}}{2}(v_{i,k} - v_{j,k}) \\ &\quad + p_{i,k} - p_{j,k} - d_{\mathrm{safe}}\ . \end{aligned} \tag{7}$$

in order to define function $G_{i,j,k}$ as

$$G_{i,j,k}(a_i, a_j, T_i, T_j, \delta_{i,j,k}) = \begin{pmatrix} \gamma_{i,j,k} - \delta_{i,j,k} - \frac{1}{4}\frac{\beta_{i,j,k}^2}{\alpha_{i,j,k} + \delta_{i,j,k}} \\ \alpha_{i,j,k} + \delta_{i,j,k} \\ \delta_{i,j,k} \end{pmatrix} \tag{8}$$

for all $(i,j) \in \mathbb{M}$ and all $k \in \mathbb{Z}_0^{K_i - 1}$.

Corollary 1 *The semi-infinite rear-end collision avoidance constraint* (4) *is satisfied if and only if there exist scalars* $\delta_{i,j,k} \in \mathbb{R}$ *such that*

$$G_{i,j,k}(a_i, a_j, T_i, T_j, \delta_{i,j,k}) > 0$$

for all $(i,j) \in \mathbb{M}$ *and all* $k \in \mathbb{Z}_0^{K_i - 1}$.

Proof. Notice that the time grid is constructed in such a way that $h_{i,k}(T_i) = h_{j,k}(T_j)$ for all $i < j$ and all $k \in \mathbb{Z}_0^{K_i-1}$. Thus, by substituting (2) we find that (4) can be written in the equivalent form

$$\begin{aligned} 0 \quad < \quad & p_{i,k} - p_{j,k} + \frac{h_{i,k}(T_i)}{2}(v_{i,k} - v_{j,k})(\tau + 1) \\ & + \frac{h_{i,k}^2(T_i)}{8}(a_{i,k} - a_{j,k})(\tau+1)^2 - d_{\mathrm{safe}} \end{aligned}$$

for all $\tau \in [0,1]$, all $(i,j) \in \mathbb{M}$, and all $k \in \mathbb{Z}_0^{N_i^1 - 1}$. Next, we can apply Lemma 1 in order to establish the statement of the corollary. □

Corollary 1 provides a computationally tractable condition, which can be used to verify that all rear-end collision avoidance constraints are satisfied. Notice that the function $G_{i,j,k}$ depends on the accelerations a_i and a_j of the i-th and j-th car and, consequently, the condition in Corollary 1 introduces a coupling between neighboring vehicles in each lane.

III. Distributed control of automuous vehicles at intersections

A. Optimal control problem formulation

This paper concerns control input optimization problems of the form

$$\begin{array}{lll} \min\limits_{a,T,\delta} & \sum_{i=1}^{M}\ J_i(a_i,T_i) & \text{OBJECTIVE} \\ \text{s.t.} & \left\{ \begin{array}{l} \forall i \in \mathbb{Z}_1^M, \forall k \in \mathbb{Z}_0^{K_i}, \\ p_{i,K_i}(a_i,T_i) = d_i^{\mathrm{in}}, \\ p_{i,N_i}(a_i,T_i) = d_i^{\mathrm{out}}, \\ 0 \leq v_{i,k}(a_i,T_i) \leq \bar{v}_i \end{array} \right. & \begin{array}{c} \text{POSITION} \\ \text{AND} \\ \text{VELOCITY} \\ \text{CONSTRAINTS} \end{array} \\ \text{s.t.} & \left\{ \begin{array}{l} \forall i \in \mathbb{Z}_1^M, \forall k \in \mathbb{Z}_0^{K_i - 1} \\ \underline{a}_i \leq a_{i,k} \leq \overline{a}_i \end{array} \right. & \begin{array}{c} \text{CONTROL} \\ \text{CONSTRAINTS} \end{array} \\ \text{s.t.} & \left\{ \begin{array}{l} \forall i \in \mathbb{Z}_1^{M-1} \\ t_i^{\mathrm{out}} \leq t_{i+1}^{\mathrm{in}} \end{array} \right. & \begin{array}{c} \text{INTERSECTION} \\ \text{COLLISION} \\ \text{AVOIDANCE} \end{array} \\ \text{s.t.} & \left\{ \begin{array}{l} \forall (i,j) \in \mathbb{M},\ \forall k \in \mathbb{Z}_0^{K_i - 1} \\ G_{i,j,k}(a_i, a_j, T_i, T_j, \delta_{i,j,k}) \geq \epsilon \end{array} \right. & \begin{array}{c} \text{REAR-END} \\ \text{COLLISION} \\ \text{AVOIDANCE} \end{array} \end{array} \tag{9}$$

for a small $\epsilon > 0$. Here, $\underline{a}_i, \overline{a}_i \in \mathbb{R}$ are given control bounds, $\overline{v}_i \geq 0$ are given speed limit of the i-th vehicle. The objective of the i-th vehicle is given by

$$\begin{aligned} J_i(a_i,T_i) := {} & Q_i \sum_{k=1}^{N_i} (v_{i,k}(a_i,T_i) - v_i^{\mathrm{ref}})^2 \\ & + R_i \sum_{k=0}^{N_i - 1} a_{i,k}^2 + S_i \sum_{k=1}^{N_i - 1} (a_{i,k} - a_{i,k-1})^2 \end{aligned} \tag{10}$$

where $v_i^{\mathrm{ref}} \in \mathbb{R}$ is a given reference velocity of the i-th vehicle, with given positive weights $Q_i, R_i, S_i > 0$. Notice that the weights R_i and S_i can be used to prioritize the comfort for the passengers in the vehicles by penalizing large accelerations as well as large changes of the acceleration.

Notice that (9) can be solved efficiently by using the distributed non-convex optimization algorithm ALADIN [11]. As elaborated in [14], ALADIN can be used to solve traffic optimization problems in a distributed way. A more detailed discussion of how ALADIN can exploit the particular band structure of matrices in (9) would go beyond the scope of this paper, but it is mentioned here that a direct application of ALADIN [11] to (9) leads to an optimization scheme, where the i-th vehicle needs to communicate with the $(i+1)$-th vehicle if $i+1 \leq M$, the $(i-1)$-th vehicle if $i-1 \geq 1$, the j_1-th vehicle if $(j_1, i) \in \mathbb{M}$, and the j_2-th vehicle if $(i, j_2) \in \mathbb{M}$. Thus, every vehicle communicates with at most 4 other vehicles; see also [11], [14] for more details.

B. Closed-loop optimal control

Algorithm 1 proposes a closed-loop control scheme for the automuous vehicle network at the traffic intersection. The main idea of this algorithm is to solve (9) whenever the vehicles have new position and velocity measurements. The steps of Algorithm 1, in particular Steps 1-3, have many similarities with a model predictive control (MPC) scheme [20], because a certainty-equivalent optimal control problem is solved at every sampling time. There are, however, two differences compared to standard MPC. Firstly, the time horizon of each vehicles is shrinking after each time step—see Step 4 in Algorithm 1. This is due to the fact that we only model the motion of the vehicles before and on the intersection area. And secondly, at each sampling time, there may be a vehicle leaving the intersection (Step 5) or there can be new vehicles arriving in the intersection area (Step 6). Thus, the number of differential states may be changing from iteration to iteration.

Algorithm 1 Distributed Closed-Loop Control Scheme

Initialization: Choose discretization accuracies $K_i \geq 0$ and $L_i > 0$ for all $i = 1, ..., M$.

Online:

1) Each vehicle measures $(\hat{p}_i, \hat{v}_i)$.
2) Use ALADIN (as in [14]) to solve (9).
3) Each vehicle applies the first control $a_{i,0}^*$ during the time interval $[0, h_{i,0}]$.
4) All vehicles set $K_i \leftarrow K_i - 1$ if $K_i > 0$. Otherwise, set $L_i \leftarrow L_i - 1$.
5) If $L_1 = 0$, the first vehicle leaves the network. This means that if there exist an index j with $(1, j) \in \mathbb{M}$, then we set
$$\mathbb{M} \leftarrow \mathbb{M} \setminus \{(1, j)\} .$$
The indices of all remaining vehicles are reset, $i \leftarrow i - 1$, $M \leftarrow M - 1$.
6) If a new vehicle arrives in the surrounding of the intersection, it asks for the index j of the vehicle in the front of its arrival lane and sets
$$\mathbb{M} \leftarrow \mathbb{M} \cup \{(j, M+1)\}$$
and then sets $M \leftarrow M + 1$.

The following lemma establishes a condition under which recursive feasibility of Algorithm 1 can be ensured.

Lemma 2 *Let the initial position $\hat{p}_{M+1}$ and velocity $\hat{v}_{M+1}$ of every incoming vehicle be such that:*

(C1) *we have*
$$\hat{p}_{M+1} + v_{\hat{M+1}} \left(\frac{\hat{v}_{M+1}}{\underline{a}_{M+1}} \right) + \frac{1}{2} \underline{a}_{M+1} \left(\frac{\hat{v}_{M+1}}{\underline{a}_{M+1}} \right)^2 \leq d_{M+1}^{\text{in}} ,$$

(C2) *if the j-th vehicle is on the same lane before the incoming vehicle, then there exist scalars δ_k and $t_{M+1}^{\text{in}}, t_{M+1}^{\text{out}} > 0$ such that*
$$G_{j,M+1,k}(a_j^*, \underline{a}_{M+1}\mathbf{1}, T_j, T_{M+1}, \delta_k) \geq \epsilon .$$
for all $k \in \mathbb{Z}_0^{K_j - 1}$. Here, a_j^* is the optimal control of the j-th vehicle that has been found in Step 2 of Algorithm 1—before the $(M+1)$-th vehicle had been arriving.

Then all optimization problems in Algorithm 1 remain recursively feasible, i.e., collisions can be avoided.

Proof. Condition (C1) ensures that—in the worst case—the new vehicle can simply stop by breaking with acceleration $\underline{a}_{M+1}$. Thus, collisions on the intersection can be avoided. Similarly, Condition (C2) ensures that if the incoming vehicle breaks with acceleration $\underline{a}_j \mathbf{1} = [\underline{a}_j, \ldots, \underline{a}_j]^{\mathsf{T}}$, then it satisfies all rear-end collision constraints with respect to the car in front of it, i.e., the solution of the previous optimal control problem can be used to construct a feasible solution of the next optimization problem. Thus, both conditions together ensure recursive feasibility of the network. □

IV. Numerical Case Study

This section illustrates the properties and performance of Algorithm 1. We use the following parameter values throughout all case studies.

TABLE I: Parameter values

d_i^{in}	0 m	d_i^{out}	10 m
d_{safe}	10 m	$\overline{v}_i$	$90\ \frac{\text{km}}{\text{h}}$
$\underline{a}_i$	$-2\ \frac{\text{m}}{\text{s}^2}$	$\overline{a}_i$	$2\ \frac{\text{m}}{\text{s}^2}$
Q	$1\ \frac{\text{s}^2}{\text{m}^2}$	R	$1\ \frac{\text{s}^4}{\text{m}^2}$
S	$1\ \frac{\text{s}^4}{\text{m}^2}$		

A. Low-traffic scenario

Let us consider a scenario with 4 vehicles and 2 lanes and with
$$\mathbb{M} = \{(1, 2), (3, 4)\} .$$

Table II summarizes the values for the initial states, references, and discretization accuracies, which are used to simulate a low-traffic scenario based on Algorithm 1. Figure 2 shows the corresponding closed-loop trajectories that are obtained for the 4 vehicles. In this scenario, one of the collision avoidance constraints in the intersection area is active. Here, vehicle 3 enters the intersection area (top-part Figure 2) immediately after vehicle 2 has left this area.

TABLE II: Parameter values for the low-traffic scenario.

	Lane 1		Lane 2	
	$i=1$	$i=2$	$i=3$	$i=4$
$\hat{p}_i$ [m]	-120	-160	-140	-180
$\hat{v}_i$ $\left[\frac{\text{km}}{\text{h}}\right]$	70	80	75	75
v_i^{ref} $\left[\frac{\text{km}}{\text{h}}\right]$	75	75	75	70
K_i	65	70	75	80
L_i	5	5	5	5

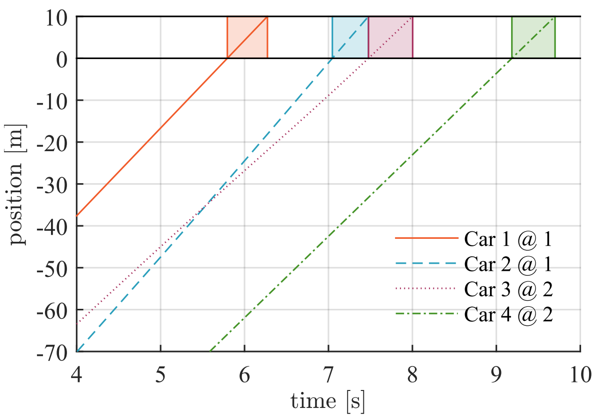

Fig. 2: Optimal position trajectories for low-traffic scenario.

In this example, all rear-end collision avoidance constraints are inactive during the whole closed-loop scenario—a situation that occurs frequently in low-traffic scenarios. This situation, however, changes in more complicated rush hour scenarios, where it becomes rather important to take rear-end collision avoidance constraints into account as shown next.

B. Rush hour scenario

Let us consider another scenario, which initially also considers 4 cars on 2 lanes. After 5 sampling times, however, a fifth vehicle enters the intersection area on a third lane. Because, there is initially no other vehicle on this third lane, the set of index pairs is—as in the previous scenario—given by

$$\mathbb{M} = \{(1,2),(3,4)\}$$

during the whole scenario. All remaining parameters for the rush hour scenario are summarized in Table III.

TABLE III: Parameter values for the rush hour scenario.

	Lane 1		Lane 2		Lane 3
	$i=1$	$i=2$	$i=3$	$i=4$	$i=5$
$\hat{p}_i$ [m]	-120	-160	-60	-75	-90
$\hat{v}_i$ $\left[\frac{\text{km}}{\text{h}}\right]$	70	80	35	56	65
v_i^{ref} $\left[\frac{\text{km}}{\text{h}}\right]$	75	75	35	33	65
K_i	65	70	75	80	80
L_i	5	5	5	5	5

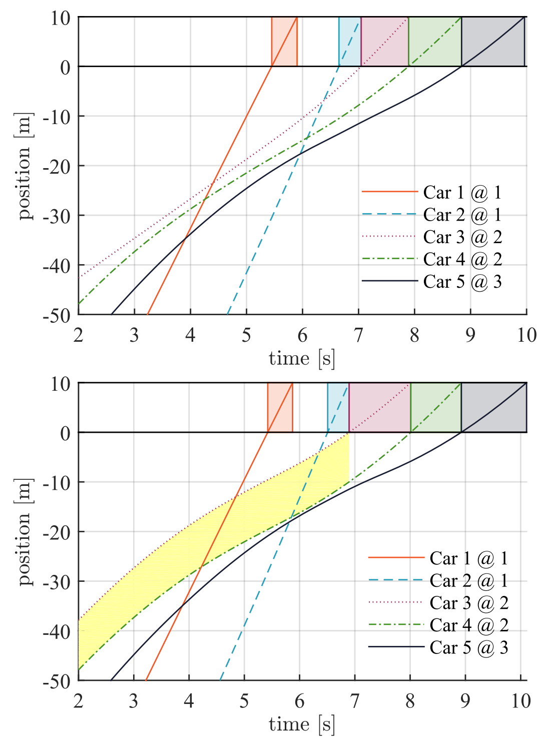

Fig. 3: The upper figure shows an optimized rush hour scenario but without enforcing rear-end collision avoidance on lane 2—in this case vehicles 3 and 4 would crash into each other. The lower figure shows the same but the collision avoidance constraints are enforced as indicated by the yellow shaded area.

Figure 3 shows the numerical results for the associated closed-loop scenario. Notice that the plug-in conditions from Lemma 2 were satisfied during this scenario, i.e., the fifth vehicle registered safely after 5 sampling times without causing infeasibilities. As indicated by the yellow shaded area in the lower part of Figure 3, the rear-end collision avoidance constraint between the vehicles on the second lane, i.e., for the pair $(3,4) \in \mathbb{M}$, is active during on the time interval $[0, t_3^{\text{in}}]$. Another interesting aspect of this case study is that the vehicles in the second lane have to wait until the vehicles from the first lane have passed the intersection area. In contrast to the behavior that could be expected from human drivers, the automuous vehicles 3 and 4 on lane 2 are much more predictive and reduce their speed much before the intersection such that they don't have to stop before the intersection area. On the contrary, one can see clearly that vehicles 3 and 4 already start accelerating again much before entering the intersection. This is optimal, because they need to pass this area quickly in order to free the intersection for the fifth vehicle on the third lane, which has registered to the network, too.

Last but not least, the proposed closed-loop algorithm based on the distributed optimization solver ALADIN has to potential to be scaled up for scenarios with more vehicles. Although a detailed discussion of such larger-scale scenarios is beyond the scope of this paper, we mention here that initial numerical tests indicate that ALADIN can easily deal with hundreds of vehicles as also discussed in [11], [14].

V. Conclusions

This paper has presented a distributed closed-loop control algorithm for autonomous vehicles at traffic intersections. We have focussed on a dynamic model for the vehicles and an associated distributed optimal control formulation that can take both collision avoidance constraints in the intersection area as well as rear-end collisions in each lane into account. In particular, Lemma 1 and Corollary 1 have introduced a reformulation strategy, which allows us to replace the semi-infinite rear-end collision avoidance constraints by numerically tractable constraints. Moreover, we have proposed a plug-and-play closed-loop control scheme, which has been summarized in Algorithm 1, and which can be used to steer all vehicles in an optimal and safe way as long as the incoming vehicles satisfy the recursive feasibility conditions from Lemma 2. Our numerical results indicate that rear-end collision avoidance constraints are especially important if one wishes to optimize the waiting behavior of the autonomous vehicles at traffic intersections during hightraffic situations. Future research will investigate more involved traffic scenarios with multiple lanes potentially allowing more than one car on the intersection at every time as long as collisions are impossible.

ACKNOWLEDGMENT

The work of Jiahe Shi, Yi Zheng, Yuning Jiang and Boris Houska was supported by the National Science Foundation China (NSFC), Nr. 61473185, as well as ShanghaiTech University, Grant-Nr. F-0203-14-012. The work of Mario Zanon and Robert Hult was supported by the Swedish Research Council (VR) under grant number 2012-4038 and the European Commission Seventh Framework under the AdaptIVe, grant agreement number 610428.

References

[1] H. Ahn, A. Colombo, and D. Del Vecchio. Supervisory control for intersection collision avoidance in the presence of uncontrolled vehicles. In *Proceedings of the 2014 American Control Conference, Portland, USA*, pages 867–873. IEEE, 2014.

[2] L. Bruni, A. Colombo, and D. Del Vecchio. Robust multi-agent collision avoidance through scheduling. In *Proceedings of the IEEE 52nd Annual Conference on Decision and Control, Florence, Italy*, pages 3944–3950. IEEE, 2013.

[3] G.R. Campos, P. Falcone, H. Wymeersch, R. Hult, and J. Sjöberg. Cooperative receding horizon conflict resolution at traffic intersections. In *Proceeding of the IEEE 53rd Annual Conference on Decision and Control*, pages 2932–2937. IEEE, 2014.

[4] L. Chen and C. Englund. Cooperative intersection management: a survey. *IEEE Transactions on Intelligent Transportation Systems*, 17(2):570–586, 2016.

[5] A. Colombo and D. Del Vecchio. Least restrictive supervisors for intersection collision avoidance: A scheduling approach. *IEEE Transactions on Automatic Control*, 60(6):1515–1527, 2015.

[6] G.R. Campos, P. Falcone, and J. Sjoberg. Autonomous cooperative driving: a velocity-based negotiation approach for intersection crossing. In *Proceeding of the 16th International IEEE Conference on Intelligent Transportation Systems*, pages 1456–1461. IEEE, 2013.

[7] K. Dresner and P. Stone. Multiagent traffic management: A reservation-based intersection control mechanism. In *Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multiagent Systems-Volume 2*, pages 530–537. IEEE Computer Society, 2004.

[8] K. Dresner and P. Stone. Multiagent traffic management: An improved intersection control mechanism. In *Proceedings of the 4th international joint conference on Autonomous agents and multiagent systems*, pages 471–477. ACM, 2005.

[9] A.L. Fradkov and V.A. Yakubovich. Thes-procedure and duality relations in nonconvex problems of quadratic programming. *Vestn. LGU, Ser. Mat., Mekh., Astron*, (1):101–109, 1979.

[10] M.R. Hafner, D. Cunningham, L. Caminiti, and D. Del Vecchio. Cooperative collision avoidance at intersections: Algorithms and experiments. *IEEE Transactions on Intelligent Transportation Systems*, 14(3):1162–1175, 2013.

[11] B. Houska, J. Frasch, and M. Diehl. An augmented lagrangian based algorithm for distributed non-convex optimization. *SIAM Journal on Optimization*, 26(2):11011127, 2016.

[12] R. Hult, G.R. Campos, P. Falcone, and H. Wymeersch. An approximate solution to the optimal coordination problem for autonomous vehicles at intersections. In *Proceeding of the 2015 American Control Conference*, pages 763–768. IEEE, 2015.

[13] R. Hult, M. Zanon, S. Gros, and P. Falcone. Primal decomposition of the optimal coordination of vehicles at traffic intersections. In *Proceeding of the IEEE 55th Conference on Decision and Control*, pages 2567–2573. IEEE, 2016.

[14] Y. Jiang, M. Zanon, R. Hult, and B. Houska. Distributed algorithm for optimal vehicle coordination at traffic intersections. In *Proceedings of the 20th IFAC World Congress, Toulouse, France*, page 1208212087, 2017.

[15] A. Katriniok, P. Kleibaum, and M. Joševski. Distributed model predictive control for intersection automation using a parallelized optimization approach. In *Proceedings of the 20th IFAC World Congress, Toulouse, France*, pages 6114–6120, 2017.

[16] D. Kouzoupis, R. Quirynen, B. Houska, M. Diehl. A block based ALADIN scheme for highly parallelizable direct optimal control. In *Proceedings of the 2016 American Control Conference*, Boston, USA, pages 1124–1129, 2016.

[17] H. Kowshik, D. Caveney, and P.R. Kumar. Provable systemwide safety in intelligent intersections. *IEEE transactions on vehicular technology*, 60(3):804–818, 2011.

[18] J. Lee and B. Park. Development and evaluation of a cooperative vehicle intersection control algorithm under the connected vehicles environment. *IEEE Transactions on Intelligent Transportation Systems*, 13(1):81–90, 2012.

[19] N. Murgovski, G.R. de Campos, and J. Sjöberg. Convex modeling of conflict resolution at traffic intersections. In *Proceeding of the IEEE 54th Annual Conference on Decision and Control*, pages 4708–4713, 2015.

[20] J.B. Rawlings and D.Q. Mayne. *Model Predictive Control: Theory and Design*. Madison, WI: Nob Hill Publishing, 2009.

[21] M. Zanon, S. Gros, P. Falcone, and H. Wymeersch. An asynchronous algorithm for optimal vehicle coordination at traffic intersections. In *Proceedings of the 20th IFAC World Congress, Toulouse, France*, 2017.