An Optimization-Based Dynamic Reordering Heuristic for Coordination of Vehicles in Mixed Traffic Intersections

Citation for the original published paper (version of record):
Faris, M., Zanon, M., Falcone, P. (2025). An Optimization-Based Dynamic Reordering Heuristic for Coordination of Vehicles in Mixed
Traffic Intersections. IEEE Transactions on Control Systems Technology, 33(4): 1387-1402.
http://dx.doi.org/10.1109/TCST.2024.3508542

(article starts on next page)

An Optimization-Based Dynamic Reordering Heuristic for Coordination of Vehicles in Mixed Traffic Intersections

Muhammad Faris, Mario Zanon, *Senior Member, IEEE*, and Paolo Falcone, *Member, IEEE*

***Abstract*— In this article, we address a coordination problem for connected and autonomous vehicles (CAVs) in mixed traffic settings with human-driven vehicles (HDVs). The main objective is to have a safe and optimal crossing order for vehicles approaching unsignalized intersections. This problem results in a mixed-integer quadratic programming (MIQP) formulation, which is unsuitable for real-time applications. Therefore, we propose a computationally tractable optimization-based heuristic that monitors platoons of CAVs and HDVs to evaluate whether alternative crossing orders can perform better. It first checks the future constraint violation that consistently occurs between pairs of platoons to determine a potential swap. Next, the costs of quadratic programming (QP) formulations associated with the current and alternative orders are compared in a depth-first branching fashion. In simulations, we show that our heuristic can be a hundred times faster than the original and simplified MIQPs (SMIQPs) and yields solutions that are close to optimal and have better order consistency.**

***Index Terms*— Autonomous vehicles (AVs), heuristic, mixed traffic, vehicle coordination.**

I. Introduction

TRAFFIC intersections, along with other merging areas such as on ramps or roundabouts, are widely recognized as significant bottlenecks in the road network, contributing to traffic-related issues that encompass both safety and inefficient use of the infrastructure. Intersection areas are prone to a considerable number of traffic accidents and fatalities [1]. To address safety concerns, strict traffic control measures are implemented, such as traffic lights and signs. Additionally, merging areas are characterized by frequent stop-and-go patterns, and the traffic flow is typically reduced by human drivers' behavior. These inefficiencies further lead to increased pollution and energy consumption [2]. Achieving traffic efficiency while ensuring safety at all times poses a significant challenge that requires innovative approaches.

Traffic lights have long been a popular conventional approach to coordinating human-driven vehicles (HDVs), using a top-down perspective [3]. However, despite their widespread use, traffic lights alone often fail to fully resolve traffic problems and achieve the desired balance between traffic efficiency and safety. One of the issues with traffic lights is their low throughput, which leads to a build-up of vehicles waiting for their turn to occupy the intersection, resulting in traffic jams. Another issue is that human errors highly contribute to accidents in signalized intersections [4].

To address these challenges and improve traffic management, connected and autonomous vehicles (CAVs) play a vital role [5]. They can leverage advanced communication and sensing technologies to unlock new possibilities for a more efficient traffic flow. By sharing real-time information and coordinating their actions through vehicle-to-everything (V2X) communication schemes, CAVs can ensure smoother traffic operations and better adherence to traffic rules than HDVs.

As research on autonomous vehicles (AVs) continues to grow, the primary focus is on gradually replacing human drivers with automation and enhancing overall traffic performance. In the future, the development of CAVs is expected to enable the implementation of unsignalized intersections, where coordination between vehicles will replace traditional traffic signals [6].

The main challenge stemming from unsignalized intersections is priority assignment, i.e., selecting a crossing order for the incoming vehicles. This problem can be formulated as a mixed-integer programming (MIP) problem and is NP-hard [7]. This renders a real-time optimal application of such problem impossible. As an alternative, first-come, first-serve (FCFS) scheduling has been widely used due to its simplicity. Unfortunately, this approach can be far from optimal [8].

Before achieving full market penetration of CAVs, transition phases will take place, where HDVs and CAVs will share the road and interact with each other. Despite a possible future high penetration of CAVs, the presence of legacy HDVs cannot be entirely ruled out. As a result, a specialized coordination strategy for CAVs is essential to effectively handle interactions with HDVs, particularly concerning occupancy at unsignalized merging areas or intersections. This dedicated strategy must ensure safe and efficient traffic flow in mixed traffic scenarios.

Accounting for the behavior of HDVs in coordination can be very complex, as decisions such as determining the crossing order and acceleration profile need continuous adaptation to account for the uncertain HDV trajectories. The presence of

Received 14 January 2024; revised 16 January 2024 and 30 July 2024; accepted 24 October 2024. Date of publication 13 December 2024; date of current version 26 June 2025. This work was supported in part by the Wallenberg Artificial Intelligence, Autonomous Systems, and Software Program (WASP) through the Knut and Alice Wallenberg Foundation. Recommended by Associate Editor F. Borrelli. *(Corresponding author: Mario Zanon.)*

Muhammad Faris is with the Department of Electrical Engineering, Chalmers University of Technology, 412 96 Gothenburg, Sweden (e-mail: farism@chalmers.se).

Mario Zanon is with the IMT School for Advanced Studies Lucca, 55100 Lucca, Italy (e-mail: mario.zanon@imtlucca.it).

Paolo Falcone is with the Department of Electrical Engineering, Chalmers University of Technology, 412 96 Gothenburg, Sweden, and also with the Dipartimento di Ingegneria "Enzo Ferrari," Universita di Modena e Reggio Emilia, 41121 Modena, Italy (e-mail: paolo.falcone@chalmers.se).

Digital Object Identifier 10.1109/TCST.2024.3508542

HDVs may trigger changes in order as the current order may become inefficient or even infeasible. Continuously monitoring and assessing the behavior of HDVs to close the control loop can be computationally challenging if the conventional approach of resolving the MIP problem is applied.

To address these issues, this work proposes a heuristic algorithm for vehicle crossing order problems in mixed traffic. The main focus is on handling the dynamic reordering of vehicles caused by changes in HDVs' predicted trajectories. The problem is first modeled and formulated as a mixed-integer quadratic programming (MIQP), where HDVs are grouped into mixed-platoons. The proposed heuristic is then derived as an approximation of the MIQP, inspired by branch-and-bound (B&B) strategies but specifically tailored to this problem. Rather than performing a standard B&B search, we exploit the knowledge of the problem to focus the expansion of the decision tree on a very small subset of all possible orders. This subset is selected based on potential future safety constraint violations, i.e., by a consistency check. At each time, an alternative crossing order that swaps two adjacent platoons is compared to the current one. The new order is retained if its cost is lower than that of the current order.

Our main contributions are the following.

1) We formulate an MIQP optimal coordination problem based on the platooning strategy.
2) We propose an optimization-based heuristic to solve the problem in a computationally tractable way.
3) We perform comprehensive numerical simulations that demonstrate the effectiveness of our approach.

This article is organized as follows. In Section II, we provide an extensive review of the literature related to the considered problem. In Section III, we define the problem and set the premises for the formulation of optimal coordination, which is provided in Section IV. In Section V, we introduce our heuristic algorithm. All simulations and discussions on the performance of the different algorithms are given in Section VI. Finally, we conclude the work in Section VII.

II. Literature Review

Early research on vehicle coordination at unsignalized intersections primarily focused on AVs. In [9], a coordination problem for AVs was proposed, suggesting the use of a safety distance between conflicting vehicles for collision avoidance. The occupancy priority was determined based on arrival times at the intersection. A similar problem was addressed by [10]. Both acknowledged the computational complexity of the crossing order problem and proposed a reachability-based heuristic method as an alternative to MIP formulation.

Several alternative heuristic methods have been utilized. The reachability-based heuristic has been further developed into a sequential priority decision-making method in [11], which aimed at reducing the number of priority permutations. In [12], a heuristic based on MIQP was introduced as an approximation of vehicle coordination using timeslots. Other trajectory optimization-based methods using constraint programming can be found in [13], [14], and [15]. To handle the complexity of NP-hard formulations, tree search methods were applied in [16] and [17] to explore possible crossing orders, finding optimal solutions through multiple directed iterations, even in lane change applications. Rule-based heuristics combined with timeslot or exit time minimization were developed in [18] and [19]. However, these heuristics are not specifically designed for mixed traffic.

More recently, researchers have been extending their focus to mixed traffic scenarios, in particular involving HDVs. Rule-based protocols were used alongside traffic lights by [3] and [20]. In [21], the same problem was addressed using the CAV-HDV platoon control method. Platooning strategies were also explored in [22] for intersection cases and in [23] for specific controlled zones. A cooperative maneuver technique was proposed by [6] for HDVs that are connected and behave according to a specific model. The crossing order was obtained through gradient-based optimization, although safety was not a primary concern for HDVs. A similar setting was considered in [24], albeit with FCFS priority. Furthermore, Faris et al. [25] studied the impact of involving HDVs on performance and safety, emphasizing that uncertainty and prediction mismatches from the HDVs may necessitate a change of order.

Some studies focus specifically on dynamic reordering or reprioritization problems. For example, Scheffe et al. [26] implemented a time-varying priority assignment by evaluating possible collisions from each vehicle. In [27], a negotiation-based priority approach was applied to coordinate CAVs, allowing rules to be negotiated during the auction phase based on the current vehicle states. Arrival/exit time minimization-based methods were used in [28], [29], and [30] to handle changing traffic flow. These methods sort the order based on individual assessments relying on conservative assumptions, such as current states only or maximum accelerations. However, they did not provide comparisons of solution quality and/or did not specifically address challenges in mixed-traffic environments.

In this article, we deal with the challenges of dynamic reordering in the context of mixed traffic. We first formulate the coordination problem as an MIQP. Then, we propose an optimization-based [quadratic programmings (QPs)] heuristic to address the computational challenge. We extensively evaluate the methods' performance in mixed traffic reordering scenarios, e.g., including when and which order to change. Moreover, comparisons with the MIQPs and simpler heuristics are provided to assess the quality of the solutions and computational tractability.

III. Problem Setup and Modeling

In this section, we introduce the problem setting and the notation used to describe both the intersection and vehicles.

A. Types of Vehicles

We consider a set of $N+M$ vehicles, where N and M are, respectively, the number of CAVs and HDVs. Each vehicle is assigned an integer index, and we denote by $\mathcal{N}$, $\mathcal{M}$ the sets of indices relative to the CAVs and HDVs, respectively.

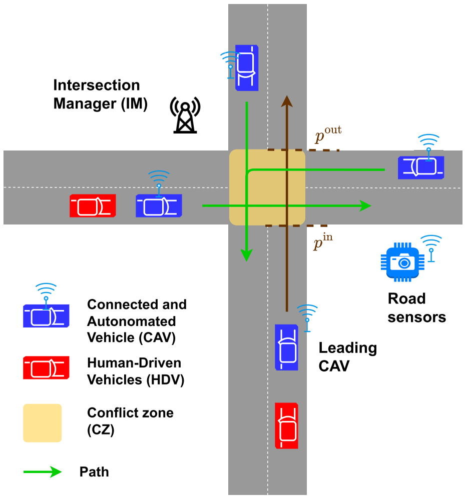
Intersection Manager (IM)
p^{out}
p^{in}
Road sensors
Leading CAV
Connected and Autonomated Vehicle (CAV)
Human-Driven Vehicles (HDV)
Conflict zone (CZ)
Path

Fig. 1. Mixed traffic of CAVs and HDVs at an intersection area.

B. Vehicle Modeling

While any vehicle model can be used within our approach, we assume for simplicity that vehicle i moves along its predefined path as described by the double-integrator model

$$x_{i,k+1} = Ax_{i,k} + Bu_{i,k} \quad \forall i \in \mathcal{N}, \mathcal{M} \tag{1}$$

where $k = \lfloor t/\Delta t \rfloor \in \mathbb{N}$ is the discrete-time index, Δt is the periodic sampling time, $t \in \mathbb{R}_+$ is time, and

$$A = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} \frac{1}{2}\Delta t^2 \\ \Delta t \end{bmatrix}.$$

The state vector $x_{i,k} = [p_{i,k},\ v_{i,k}]^\top$ contains the longitudinal distance of vehicle i from its origin point and its velocity. Each vehicle starts at $k = 0$ with given initial states

$$x_{i,0} = x_i^0 \quad \forall i \in \mathcal{N}, \mathcal{M} \tag{2}$$

and is subject to the following velocity and acceleration/deceleration (input) bounds:

$$v^{\min} \le v_{i,k} \le v^{\max} \quad \forall i \in \mathcal{N}, \mathcal{M} \tag{3a}$$
$$u^{\min} \le u_{i,k} \le u^{\max} \quad \forall i \in \mathcal{N}, \mathcal{M} \tag{3b}$$

with $v^{\min} > 0$, as we assume that vehicles cannot reverse.

Based on the concept discussed in [25], the HDVs' behavior is described by the following mix of constant-velocity and maximum acceleration/deceleration models:

$$v_{i,k+1} = v_{i,k} + \Delta t u_{i,k}, \quad k > 0 \quad \forall\ i \in \mathcal{M} \tag{4}$$

where

$$u_{i,k} = \begin{cases} 0, & u_{i,k-1} \ge 0 \\ u^{\min}, & u_{i,k-1} < 0 \end{cases}$$

in which $u^{\min}$ is the deceleration limit. Note that, similar to [31] and [32], we chose the model above for the sake of simplicity due to the assumption on the limited information on HDVs' behavior. Alternative HDV models exist and can be used instead of the one we propose without any negative impact on our method.

C. Conflict Zones

We call a conflict zone (CZ) any portion of an intersection area where vehicles coming from different directions intersect their paths, as illustrated in Fig. 1.

For simplicity, we consider the case of a single CZ at the center of the intersection area. The CZ is defined by the pairs $(p^{\text{in}}, p^{\text{out}})$ denoting the entry and exit positions in each direction, along each path. While having a single CZ here can be conservative, we restricted to that setting for the sake of simplicity. An extension to multiple CZs as in, e.g., [33], will be considered in future work. Furthermore, for simplicity, we consider the case of one lane per direction, which implies that no overtaking is allowed between vehicles coming from the same direction. To avoid lateral (side) collisions due to conflicting paths, each vehicle must occupy the CZ exclusively as we will discuss in more detail in Section IV.

D. Platoon Roles

In this article, we aim to exploit the presence of CAVs to efficiently regulate the traffic at the intersection area. The main idea is that, by adapting the speed of CAVs approaching the intersection, the behavior of HDVs can be influenced to optimize the overall intersection efficiency and safety.

As stated in Assumption 1, an autonomous intersection manager (IM), i.e., a coordinator is present [34], [35], which assigns a CAV i the role of *platoon leader* if it approaches the intersection ahead of at least one HDV. In a practical implementation, CAVs can coordinate in a fully distributed setting to handle these tasks instead so that an IM is not required. However, this requires extensions to the current algorithm, which will be the subject of future work. Note also that, with the advancement of wireless communication techniques (5G, 6G), the IM need not be located in the intersection but can be in the cloud.

We denote the last vehicle in each platoon i as *tail* m. The platoon length $l_{i,k}$ is defined as the position difference between the positions of the platoon leader and tail vehicles

$$l_{i,k} = p_{i,k} - p_{m,k}. \tag{5}$$

Additionally, we assume that IM determines the member of the platoon a priori based on, e.g., an inter-vehicle distance rule. Additional rules can also be introduced to allow for further flexibility such as, e.g., splitting a long platoon in case the IM deems it useful. Finally, a CAV that is not followed by any HDV forms a *one-vehicle platoon*, and an HDV that is not preceded by any CAV is designated as a leader of itself.

As HDVs do not have connectivity, they are not assigned an active role.

Assumption 1: HDVs cannot communicate with other vehicles or the IM. Nevertheless, their current states and inputs can be measured by the road infrastructure and are available to the IM. Moreover, a platoon must remain intact when crossing the intersection, i.e., no vehicle coming from other directions can divide the platoon.

Finally, we assume that a high penetration of CAVs is achieved, which we state as $N \gg M$. This reduces the

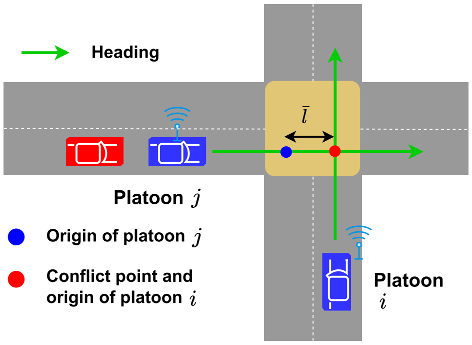

Fig. 2. Change of coordinate.

probability of having more leading HDVs coming to the intersection that is not preceded by any CAV, which can potentially induce complicated situations. Note that this assumption is not necessary for the theory, but just to guarantee that our approach can yield some performance increase.

IV. Coordination Problem

A. Intersection Crossing Order

Let $\mathcal{O}_k = [o_{1,k}, \ldots, o_{n,k}, \ldots, o_{N,k}]^\top$ be a vector containing the CAVs' crossing orders at time k, where $o_{n,k} = i \in \mathcal{N}$ if at time k CAV i is planned to cross at the nth position. Note that only the leading CAVs are ordered in $\mathcal{O}_k$, while the HDVs inherit the order assigned to the CAV leading their platoon.

As for the non-cooperative *leading* HDVs, they are not assigned any order and each of them is considered as a one-vehicle platoon. In order to guarantee safety, the IM can impose additional safety constraints with the aim of avoiding collisions, e.g., as in (14).

B. Safety Constraints

In the considered problem setup, the vehicles approaching the intersection must avoid two types of collisions: lateral (side) collisions, which can occur in the CZ; and rear-end collisions, which can occur inside a platoon or between platoons that move in the same direction.

1) Lateral Collision Avoidance: To avoid side collisions within the CZ, the position gap between two platoons $i, j \in \mathcal{N}$ that arrive from different directions must be greater than a given constant $d^{\min}$. This collision avoidance condition is formulated within the following conditional constraint, which also defines the crossing order:

$$\text{if } t_j^{\text{out}} \le t_i^{\text{in}} : p_{j,k} \ge p_{i,k} + d^{\min} + \bar{l}, \quad k \in \left[\underline{k}_j, \bar{k}_i\right] \tag{6a}$$

$$\text{if } t_i^{\text{out}} \le t_j^{\text{in}} : p_{i,k} \ge p_{j,k} + d^{\min} + \bar{l}, \quad k \in \left[\underline{k}_i, \bar{k}_j\right] \tag{6b}$$

where $\underline{k}_c = \min(k \mid t_k \ge t_c^{\text{in}})$, $\bar{k}_c = \max(k \mid t_k \le t_c^{\text{out}})$, $c \in \{i, j\}$ and $t_k = k\Delta t$. Constant $\bar{l}$ accounts for the change of coordinates between platoons i and j, as illustrated in Fig. 2 and inspired by the *virtual platooning* concept [36]. Note that either (6a) or (6b) is active w.r.t. the selected sequence, i.e., $t_j^{\text{out}} \le t_i^{\text{in}}$ means j enters CZ before i and vice versa within certain timeslots, i.e., $[\underline{k}_j, \bar{k}_i] \ \vee \ [\underline{k}_i, \bar{k}_j]$, respectively.

The following conditions relate the timeslots to the platoons i, j and CZ positions:

$$p_{c,\underline{k}_c} \ge p^{\text{in}}, \quad p_{c,\bar{k}_c} \le p^{\text{out}} \tag{7a}$$

$$p_c\left(t_c^{\text{in}}\right) = p^{\text{in}}, \quad p_c\left(t_c^{\text{out}}\right) = p^{\text{out}}. \tag{7b}$$

As we consider platoons as single vehicles (Assumption 1), this constraint must account for the *platoon length* $l_{i,k}$.

Accordingly, binary indicators $\rho^{\text{in}}_{i,j,k}, \rho^{\text{out}}_{i,j,k} \in \{0, 1\}$ are introduced to activate the constraint within the selected timeslots, which rewrites the constraint as follows:

$$\left(\rho^{\text{in}}_{i,j,k} - \rho^{\text{out}}_{i,j,k}\right)\left(1 - r_{i,j}\right)\left(p_{j,k} - l_{j,k} - p_{i,k} - d^{\min} - \bar{l}\right) \ge 0 \tag{8a}$$

$$\left(\rho^{\text{in}}_{i,j,k} - \rho^{\text{out}}_{i,j,k}\right)\left(r_{i,j}\right)\left(p_{i,k} - l_{i,k} - p_{j,k} - d^{\min} - \bar{l}\right) \ge 0 \tag{8b}$$

where $r_{i,j} \in \{0, 1\}$ is a binary variable defining whether i crosses before j ($r_{i,j} = 1$) or the converse ($r_{i,j} = 0$).

The values of $\rho^{\text{in}}_{i,j,k}, \rho^{\text{out}}_{i,j,k}$ depend on the times at which the platoons occupy the CZ, i.e., they must satisfy these conditions

$$\rho^{\text{in}}_{i,j,k} \le 1 + \frac{\bar{p}_k^{\text{in}} - p^{\text{in}}}{M^b} \tag{9a}$$

$$\rho^{\text{in}}_{i,j,k} \ge \frac{\bar{p}_k^{\text{in}} - p^{\text{in}}}{M^b} \tag{9b}$$

$$\rho^{\text{out}}_{i,j,k} \le 1 + \frac{\bar{p}_k^{\text{out}} - p^{\text{out}}}{M^b} \tag{9c}$$

$$\rho^{\text{out}}_{i,j,k} \ge \frac{\bar{p}_k^{\text{out}} - p^{\text{out}}}{M^b} \tag{9d}$$

where

$$\bar{p}_k^{\text{in}} = r_{i,j} p_{i,k} + \left(1 - r_{i,j}\right) p_{j,k} \tag{10}$$

$$\bar{p}_k^{\text{out}} = r_{i,j} p_{j,k} + \left(1 - r_{i,j}\right) p_{i,k} \tag{11}$$

and M^b is a sufficiently large constant value, i.e., Big-M [37]. Conditions (9a) and (9b) are used to set $\rho^{\text{in}}_{i,j,k}$ to 0 when *either* of the platoons i, j (OR condition) is before p^{in} and to 1 otherwise. Similarly, conditions (9c) and (9d) are used to set $\rho^{\text{out}}_{i,j,k}$ to 0 or 1, respectively, when *both* platoons (AND condition) are before or after p^{out}. The implication of these conditions to constraint (8) is illustrated in Fig. 3, which displays the situation in which platoon i reaches the intersection before j, such that the constraint (8) becomes active first. After this time, we have $\rho^{\text{in}}_{i,j,k} = 1$. Similarly, after both vehicles have exited the intersection, $\rho^{\text{out}}_{i,j,k} = 1$, and the collision avoidance constraint is not enforced anymore.

Avoiding the multiplication of integer variables in (8) is convenient from a coordination problem formulation standpoint [37], as shown next. Indeed, the conditions (8)–(11) can be rewritten as follows:

$$M^b\left(1 - \rho^{\text{in}}_{i,j,k} + \rho^{\text{out}}_{i,j,k} + 1 - r_{i,j}\right) + p_{j,k} - l_{j,k} - p_{i,k} - d^{\min} - \bar{l} \ge 0 \tag{12a}$$

$$M^b\left(1 - \rho^{\text{in}}_{i,j,k} + \rho^{\text{out}}_{i,j,k} + r_{i,j}\right) + p_{i,k} - l_{i,k} - p_{j,k} - d^{\min} - \bar{l} \ge 0 \tag{12b}$$

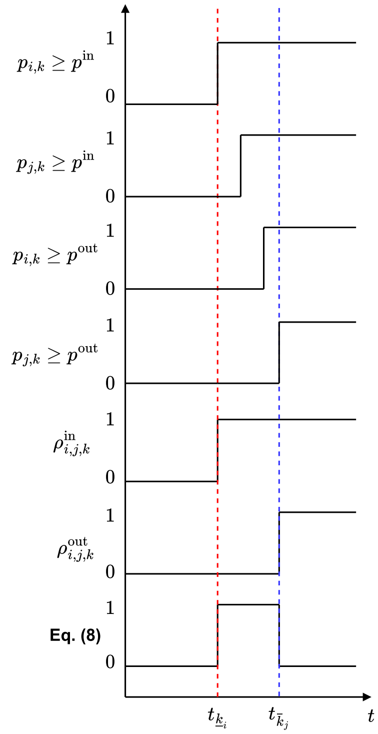

Fig. 3. Timing diagram of the safety constraint (8) activation/deactivation, where the entry condition is based on OR logic of a pair of platoons i, j positions w.r.t. CZ, while the exit condition is based on AND.

along with the position constraints (9), (11)

$$p_{i,k} - p^{\text{in}} \leq M^b \rho^{\text{in}}_{i,j,k} \tag{13a}$$
$$p_{j,k} - p^{\text{in}} \leq M^b \rho^{\text{in}}_{i,j,k} \tag{13b}$$
$$-2M^b (1 - r_{i,j}) - p_{i,k} + p^{\text{in}} \leq -M^b \left(\rho^{\text{in}}_{i,j,k} - 1\right) \tag{13c}$$
$$-2M^b (r_{i,j}) - p_{j,k} + p^{\text{in}} \leq -M^b \left(\rho^{\text{in}}_{i,j,k} - 1\right) \tag{13d}$$
$$-2M^b (r_{i,j}) + p_{i,k} - p^{\text{out}} \leq M^b \rho^{\text{out}}_{i,j,k} \tag{13e}$$
$$-2M^b (1 - r_{i,j}) + p_{j,k} - p^{\text{out}} \leq M^b \rho^{\text{out}}_{i,j,k} \tag{13f}$$
$$-p_{i,k} + p^{\text{out}} \leq -M^b \left(\rho^{\text{out}}_{i,j,k} - 1\right) \tag{13g}$$
$$-p_{j,k} + p^{\text{out}} \leq -M^b \left(\rho^{\text{out}}_{i,j,k} - 1\right). \tag{13h}$$

It can be verified that (13a) and (13b) imply (9b), while (13c) and (13d) correspond to (9a). Similarly, (13g) and (13h) imply (9d), while (13g) and (13h) correspond to (9c).

To maintain safety in the case of leading HDVs, the IM can impose a constraint similar to (6) to force the CAV-led platoons to let the leading HDVs occupy the intersection first, i.e.,

$$\left(\rho^{\text{in}}_{i,j,k} - \rho^{\text{out}}_{i,j,k}\right) \left(p_{j,k} - p_{i,k} - d^{\min} - \bar{l}\right) \geq 0 \tag{14}$$

where i, j are the indices of the CAV-led platoon and the leading HDV, respectively. Variables $\rho^{\text{in}}_{i,j,k}, \rho^{\text{out}}_{i,j,k}$ are determined according to (13).

2) Rear-End Collision Avoidance: Consider two adjacent vehicles coming from the same direction, with platoon i behind platoon j. To avoid rear-end collisions, the position gap between the two platoons must be no smaller than $d^{\min}$

$$p_{j,k} - l_{j,k} - p_{i,k} \geq d^{\min}. \tag{15}$$

C. Objective Function

A CAV $i \in \mathcal{N}$ aims to follow its reference speed profile $v^{\text{ref}}_{i,k}$ while also minimizing its acceleration/deceleration over some *prediction* horizon $K^{\text{pre}} \in \mathbb{N}$ (or $T^{\text{pre}} \in \mathbb{R}_+$). This can be formalized by the cost function

$$J_i = \sum_{k=\lfloor \frac{t}{\Delta t} \rfloor}^{K^{\text{pre}}-1} q^v \left(v^{\text{ref}}_{i,k} - v_{i,k}\right)^2 + q^u u^2_{i,k} + q^v \left(v^{\text{ref}}_{i,k} - v_{i,k}\right)^2 \tag{16}$$

where q^v and q^u are constant weights and

$$v^{\text{ref}}_{i,k} = \begin{cases} v_{m,k}, & l_{i,k-1} \geq \bar{d} \\ v^{\text{nom}}, & l_{i,k-1} < \bar{d} \end{cases} \tag{17}$$

where m is the index of the HDV following the *leading* CAV i and $v^{\text{nom}} \geq v_{m,k}$ is the ith CAV's preferred speed. The reference speed in (17) aims at limiting the platoon length $l_{i,k-1}$ in case the following mth HDV is falling behind and is set to be constant over K^{pre} steps.

D. Problem Formulation

The vehicle coordination problem can be formulated at the time t as the following MIQP constrained optimal control problem:

$$\Phi^{\text{OMIQP}}\left(\mathbf{v}^{\text{ref}}_k, \boldsymbol{x}_0, \mathbf{p}^h\right) = \min_{\mathbf{r}, \boldsymbol{\rho}, \mathbf{w}} \sum_i^N J_i\left(\mathbf{w}_i\right) \tag{18a}$$
$$\text{s.t. } x_{i,k+1} = A x_{i,k} + B u_{i,k} \tag{18b}$$
$$x_{i,0} = x^0_i, \tag{18c}$$
$$v^{\min} \leq v_{i,k} \leq v^{\max} \tag{18d}$$
$$u^{\min} \leq u_{i,k} \leq u^{\max} \tag{18e}$$
$$\text{Eq. (12), (13), (15)} \tag{18f}$$
$$r_{i,j}, \rho^{\text{in}}_{i,j,k}, \rho^{\text{out}}_{i,j,k} \in \{0, 1\} \tag{18g}$$

where $\mathbf{p}^h = [\mathbf{p}_{i,0:K^{\text{pre}}}, \ldots, \mathbf{p}_{M,0:K^{\text{pre}}}]^\top$ collects the predicted trajectories of *tail* HDVs, which define the platoon length $l^p_{i,k}$ appearing in (12) and (15), $\boldsymbol{x}_0 = [x_{1,0} \ldots, x_{N,0}]^\top$, collects the initial states of all CAVs, $\mathbf{r} = [r_{1,2}, \ldots, r_{i,j}, \ldots, r_{N-1,N}]^\top$, $\boldsymbol{\rho}^{\text{in}} = [\rho^{\text{in}}_{1,2,0}, \ldots, \rho^{\text{in}}_{N-1,N,K^{\text{pre}}}]^\top$, $\boldsymbol{\rho}^{\text{out}} = [\rho^{\text{out}}_{1,2,0}, \ldots, \rho^{\text{out}}_{N-1,N,K^{\text{pre}}}]^\top$, $\boldsymbol{\rho} = [(\boldsymbol{\rho}^{\text{in}})^\top, (\boldsymbol{\rho}^{\text{out}})^\top]^\top$ collects the binary variables encoding the crossing order and safety constraints timing. Furthermore, $\mathbf{v}^{\text{ref}}_k = [v^{\text{ref}}_{1,k}, \ldots, v^{\text{ref}}_{N,k}]^\top$ collects the reference velocities, and the continuous optimization variables are lumped in $\mathbf{w} = [\mathbf{w}_1, \ldots, \mathbf{w}_i, \ldots, \mathbf{w}_N]^\top$, where $\mathbf{w}_i = [w_{i,1}, \ldots, w_{i,k}, \ldots, w_{i,K^{\text{pre}}}]^\top \in \mathbf{R}^{n^{\text{w}} \times 1}$ with $w_{i,k} = [x_{i,k}, u_{i,k}]^\top$ lumping together the states and control inputs of vehicle i. Note that the number of platoons that are scheduled to cross after platoon $i \in \{i, N-1\}$ can be obtained as $\sum_j r_{i,j}$, $\forall j \in \{i+1, N\}$, and the crossing order $\mathcal{O}_k$ can be constructed from $\mathbf{r}$ directly.

When solving MIQP (18) above, by setting $r_{i,j}$ to either 0 or 1, the solver selects which of the two (complementary) conditions in (12) will be enforced to define the optimal order.

To avoid infeasibility in case the initial position difference between platoons is lower than $d^{\min}$, a slack variable $\eta_{i,j,k} \geq 0$ is added to the equations

$$\begin{aligned} &M^b\left(1-\rho_{i,j,k}^{\text{out}}+\rho_{i,j,k}^{\text{in}}+1-r_{i,j}\right) \\ &\quad +p_{j,k}-l_{j,k}-p_{i,k}-d^{\min}-\bar{l}+\eta_{i,j,k} \geq 0 \qquad (19a) \\ &M^b\left(1-\rho_{i,j,k}^{\text{out}}+\rho_{i,j,k}^{\text{in}}+r_{i,j}\right) \\ &\quad +p_{i,k}-l_{i,k}-p_{j,k}-d^{\min}-\bar{l}+\eta_{i,j,k} \geq 0 \qquad (19b) \end{aligned}$$

and a linear term is introduced in the cost to penalize this relaxation so that Problem (18) is reformulated as follows:

$$\begin{aligned} &\Phi^{\text{OMIQP}}\left(\mathbf{v}_k^{\text{ref}}, \boldsymbol{x}_0, \mathbf{p}^h\right) = \\ &\min_{\mathbf{r},\boldsymbol{\rho},\mathbf{w},\boldsymbol{\eta}} \sum_{i}^{N} J_i\left(\mathbf{w}_i\right)+\sum_{i=1}^{N-1}\sum_{j=i+1}^{N} J^e\left(\boldsymbol{\eta}_{i,j}\right) \qquad (20a) \\ &\text{s.t. } x_{i,k+1}=Ax_{i,k}+Bu_{i,k} \qquad (20b) \\ &\quad x_{i,0}=x_i^0 \qquad (20c) \\ &\quad v^{\min} \leq v_{i,k} \leq v^{\max} \qquad (20d) \\ &\quad u^{\min} \leq u_{i,k} \leq u^{\max} \qquad (20e) \\ &\quad \text{Eq. (19), (13), (15)} \qquad (20f) \\ &\quad r_{i,j}, \rho_{i,j,k}^{\text{in}}, \rho_{i,j,k}^{\text{out}} \in \{0,1\} \qquad (20g) \end{aligned}$$

where

$$J^e=\sum_{k=\lfloor \frac{t}{\Delta t}\rfloor}^{K^p-1} q^{e,l}\eta_{i,j,k}+\sum_{k=\lfloor \frac{t}{\Delta t}\rfloor}^{K^p-1} q^{e,q}\left(\eta_{i,j,k}\right)^2. \qquad (21)$$

Variable $\boldsymbol{\eta}_{i,j}=\left[\eta_{i,j,0} \ldots \eta_{i,j,K^{\text{pre}}}\right]$ collects the slacks for each pair $i \in \{1, N-1\}, j \in \{i+1, N\},\ i \neq j$ for all k, and $q^{e,l}, q^{e,q}$ are constants weights for linear and quadratic terms, respectively. The quadratic term is, in principle, not needed, but we add it to the cost to introduce some positive curvature that can help the solver converge faster. From this point on, we denote MIQP (20) as the original MIQP (OMIQP).

The OMIQP is executed in a closed-loop fashion according to Algorithm 1, lines 4–5. With respect to a standard closed-loop implementation, in this case, we need to account for the fact that vehicles will eventually reach the intersection's entry, such that the definition of an order will eventually become meaningless. To address that issue, we apply a simple strategy of not updating the crossing order once the vehicles are too close to the intersection. Accordingly, the fixed-order counterpart of MIQP (20), given in (25), is solved on lines 7–8 to keep yielding CAVs' control input.

In the context of continuous traffic flow, our algorithm can be readily modified to take into account the incoming platoons and subsequently coordinate their order and trajectories. Once a platoon reaches the entry, we can fix its sequence and exclude it from the MIQP problem. For simplicity, we do not consider this scheme for now and instead use a static number of platoons.

E. Lower-Complexity/SMIQP

Solving the OMIQP (20) which imposes the safety constraints (19) in a closed-loop way can be computationally heavy as the timing binaries $\boldsymbol{\rho}$ are strictly upper- and lower-bounded at each time k, as shown by conditions (13).

Algorithm 1 Original MIQP

```
Input: $\mathbf{v}_k^{\text{ref}}, \boldsymbol{x}_0, \mathbf{p}^{\text{h}}$
Output: $\mathcal{O}_k$, w
 1: for $k \in \mathbb{R}^+$ do
 2:     Obtain $\mathbf{v}_k^{\text{ref}}, \boldsymbol{x}_0, \mathbf{p}^{\text{h}}$          ▷ Parameter
 3:     if $p_{i,k} \leq p^{\text{in}},\ \forall i \in \mathcal{M}, \mathcal{N}$ then
 4:         Solve OMIQP $\Phi^{\text{U}}$ (20) / (23)
 5:         Obtain $\mathcal{O}_k$ from r
 6:     else
 7:         Set $\mathcal{O}_k = \mathcal{O}_{k-1}$ and obtain r
 8:         Solve fixed-order QP (25)
 9:     end if
10:     Apply w to CAVs of platoons
11: end for
```

However, the mechanism illustrated in Fig. 3 can still be realized without (13c)–(13f). This is because they are used to deactivate constraint (19), i.e., by setting $\rho_{i,j,k}^{\text{in}}=0$ and $\rho_{i,j,k}^{\text{out}}=1$, which implies less restriction on the solution space and potentially produces solutions with lower costs. Therefore, the solver will try to achieve it without the presence of (13c)–(13f) anyway.

Accordingly, we can propose the following simplified conditions:

$$\begin{aligned} p_{i,k}-p^{\text{in}} &\leq M^b\rho_{i,j,k}^{\text{in}} &(22a) \\ p_{j,k}-p^{\text{in}} &\leq M^b\rho_{i,j,k}^{\text{in}} &(22b) \\ -p_{i,k}+p^{\text{out}} &\leq -M^b\left(\rho_{i,j,k}^{\text{out}}-1\right) &(22c) \\ -p_{j,k}+p^{\text{out}} &\leq -M^b\left(\rho_{i,j,k}^{\text{out}}-1\right) &(22d) \\ \rho_{i,j,k}^{\text{in}} &\leq \rho_{i,j,k+1}^{\text{in}} &(22e) \\ \rho_{i,j,k}^{\text{out}} &\leq \rho_{i,j,k+1}^{\text{out}}. &(22f) \end{aligned}$$

Additionally, we introduce less complex conditions (22e) and (22f) to prevent the activation of (19) at time k before $k+1$ is active by exploiting the fact that $v^{\min}>0$, i.e., vehicles are closer to the intersection in each time t.

As we will demonstrate in the simulations part, i.e., Section VI, this simplification yields an MIQP that can be solved much faster than the original one and converges within the imposed solver iterations limit. The simplified MIQP (SMIQP) is formulated as follows:

$$\begin{aligned} &\Phi^{\text{SMIQP}}\left(\mathbf{v}_k^{\text{ref}}, \boldsymbol{x}_0, \mathbf{p}^h\right) \\ &=\min_{\mathbf{r},\boldsymbol{\rho},\mathbf{w},\boldsymbol{\eta}} \sum_{i}^{N} J_i\left(\mathbf{w}_i\right)+\sum_{i=1}^{N-1}\sum_{j=2}^{N} J^e\left(\boldsymbol{\eta}_{i,j}\right) \qquad (23a) \\ &\text{s.t. } x_{i,k+1}=Ax_{i,k}+Bu_{i,k} \qquad (23b) \\ &\quad x_{i,0}=x_i^0 \qquad (23c) \\ &\quad v^{\min} \leq v_{i,k} \leq v^{\max} \qquad (23d) \\ &\quad u^{\min} \leq u_{i,k} \leq u^{\max} \qquad (23e) \\ &\quad \text{Eq. (19), (22), (15)} \qquad (23f) \\ &\quad r_{i,j}, \rho_{i,j,k}^{\text{in}}, \rho_{i,j,k}^{\text{out}} \in \{0,1\}. \qquad (23g) \end{aligned}$$

As this problem is an alternative to MIQP (20), we will also adopt the closed-loop strategy described in Algorithm 1.

V. Heuristic Approach

As mentioned before, we aim to develop a heuristic algorithm to avoid solving MIQP (20) or (23) for closed-loop applications. As we will show, this allows us to significantly reduce the computational burden.

The heuristic we propose exploits the structure of the problem. In particular, if the current crossing order $\mathcal{O}_k$ is either leading to a potential accident or is no longer beneficial, a safe coordination can be obtained by selecting a different order.

At every time step, we monitor the safety constraints to check whether they will be violated or not in the future. If any platoon consistently violates the constraint, we try to swap its order with an adjacent platoon; we compare the cost of the two (current and alternative) orders; and we retain the order with the lowest cost. To evaluate multiple alternative orders, a depth-first branching strategy [37] as illustrated in Fig. 4 is applied for the cost comparison.

The complete pseudocode for the proposed heuristic is given in Algorithm 2, which we explain in detail next.

A. One-Time MIQP

During initialization, we obtain the first crossing order $\mathcal{O}_k = 0$ by solving the MIQP (23), which is then designated as the *initial* current order $\mathcal{O}_0$, see line 3 of Algorithm 2. The MIQP is solved only once in the beginning so that the order used by the platoons to start with is guaranteed to be both feasible and optimal. Also, the resulting position trajectories are used to approximate timing parameter $\hat{\boldsymbol{\rho}}$, which we then use to solve the QP (25) for the next time t. This will be discussed in detail in Section V-C.

As an alternative to solving the MIQP, one may opt for heuristic methods, such as, e.g., FCFS or predicted arrival time to intersection, to reduce the computational burden. Future research will aim at finding alternative approaches to address this problem.

B. Consistency Check

A change of order may be necessary when the position gap between two adjacent platoons becomes too small. To that end, we perform the following consistency check.

1) Step 1: At each time step $k > 0$, we check the collision avoidance constraints (12)–(13) over K^{pre} for each pair of platoons i, j (see lines 8–16 in Algorithm 2). In particular, this check is applied to any pair whose *preceding* one is (currently) a *mixed* platoon, as it contains HDV(s) which may considerably change their trajectories. Hence, the following platoons are collected in the set $\mathcal{N}_k^T$ (lines 4 and 23).

To perform the check, we construct the vector

$$h_i^s = \begin{bmatrix} d^{\min} - p_{j,0} + l_{j,0} + p_{i,0} + \bar{l}, \\ \vdots \\ d^{\min} - p_{j,k} + l_{j,k} + p_{i,k} + \bar{l}, \\ \vdots \\ d^{\min} - p_{j,K^{\text{pre}}} + l_{j,K^{\text{pre}}} + p_{i,K^{\text{pre}}} + \bar{l} \end{bmatrix} \quad \forall i \in \mathcal{N}_k^{\text{T}} \tag{24}$$

where $l_{j,k}$ are computed from predicted HDV trajectories using (4) (line 6) and the optimal position of the CAVs $p_{i,k}, p_{j,k}$ are obtained from the solution of (25) at the previous time $k-1$ (line 9).

2) Step 2: A (predicted) violation is obtained if any of the components of h_i^s becomes positive. We keep track of how many constraint violations occur by the scalar s_i (line 11). We use this variable to trigger a potential reorder whenever $s_i = n^{\max}$, where $n^{\max}$ is a fixed parameter (line 13).

Setting $n^{\max} > 1$ allows us to robustify against false positive triggers, i.e., a reordering that is then followed by another reordering that restores the original order, which might be due to factors, e.g., noise, and can cause chattering behavior.

3) Step 3: Whenever $s_i = n^{\max}$ for some i, we add i to set $\mathcal{E}_k$ (lines 13–15). This set will be used next to restrict the reordering procedure to only consider potential swaps between platoon i and the preceding one j, i.e., only $i \in \mathcal{E}_k$ will be considered in the cost comparison step explained next.

C. Cost Comparison

The reordering procedure is guided by the set $\mathcal{E}_k$, such that we only consider specific sets of platoons that have been consistently violating (24) (Algorithm 2, line 17). For each platoon $i \in \mathcal{E}_k$, the algorithm compares the cost of the current and alternative (swapped) order, given the current measurements/parameter at time t_k.

1) *Subproblem A*: Solve OCP (20)/(23) with the fixed, *current* order, i.e., platoon i follows the platoon j. We introduce the order with this fixed sequence as $O_{k-1}^{i|j}$, with the superscript notation here indicating the specific sequences of i, j. The rest of the sequences in $O_{k-1}^{i|j}$ are copied from the previous crossing order and left unchanged. As the order is fixed, the binaries $\mathbf{r}^{i|j}$ are also fixed and become parameters. To further reduce computational complexity, the timing binaries $\hat{\boldsymbol{\rho}}$ are approximated by using the predicted trajectory $\mathbf{w}$ from the previous time $k-1$ (lines 3 and 29). Since all the binary decision variables have become parameters, we obtain the following FOQP:

$$\Phi_{i|j}^{\text{FOQP}}\left(\mathbf{v}_k^{\text{ref}}, \boldsymbol{x}_0, \mathbf{p}^h, \mathbf{r}^{i|j}, \hat{\boldsymbol{\rho}}\right) \tag{25a}$$

$$= \min_{\mathbf{w}, \boldsymbol{\eta}} \sum_{i}^{N} J_i\left(\mathbf{w}_i\right) + \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} J^e\left(\boldsymbol{\eta}_{i,j}\right) \tag{25b}$$

$$\text{s.t. } x_{i,k+1} = A x_{i,k} + B u_{i,k} \tag{25c}$$

$$x_{i,0} = x_i^0 \tag{25d}$$

$$v^{\min} \le v_{i,k} \le v^{\max} \tag{25e}$$

$$u^{\min} \le u_{i,k} \le u^{\max} \tag{25f}$$

$$\text{Eq. (19), (15).} \tag{25g}$$

2) *Subproblem B*: The sequence of the pair i, j is *reversed* in $\mathcal{O}_k$, such that CAV j follows CAV i. This defines an *alternative* order $O_{k-1}^{j|i}$. Using this new order, we solve FOQP (25) $\Phi_{j|i}^{\text{FOQP}}(\mathbf{v}_k^{\text{ref}}, \boldsymbol{x}_0, \mathbf{p}^h, \mathbf{r}^{j|i}, \hat{\boldsymbol{\rho}})$, where the

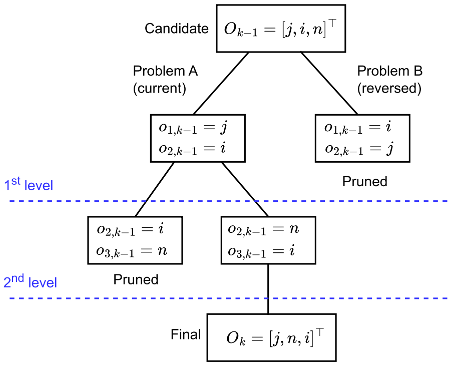

Fig. 4. Branching process of the current and alternative (reversed) subproblems performed in the proposed heuristic.

difference here is that we replace $\mathbf{r}^{i|j}$ with $\mathbf{r}^{j|i}$ to account for the different order. Note that the same timing binaries as in Subproblem A are applied here. Consequently, the lateral collision avoidance constraint will only be enforced approximately. In practice, this does not create safety concerns, as large safety margins need to be introduced anyway by allowing the change of order only before the platoons are close to the intersection; Also, the cost estimate is only approximate.

Once both subproblems are solved, we compare their costs on lines 18–19. If there exists more than one platoon in $\mathcal{E}_k$, a depth-first sequential branching for the cost comparison here is performed, as illustrated in Fig. 4. Thus, the next *current* crossing order might be updated at each iteration (level) of the branching (line 20). Finally, the new order $\mathcal{O}_k$ will be the one associated with the subproblem yielding the lowest cost.

Note that, if any change of order (line 22) takes place at time t_k, the set that contains the platoons that immediately comes after the mixed-platoons $\mathcal{N}_k^{\mathrm{T}}$ needs to be updated (lines 23).

Furthermore, as previously in Algorithm 1, the order is allowed to change as long as no platoon is close to the CZ (line 7), otherwise it is frozen. The current order is kept when the order is frozen or $\mathcal{E}_k$ is empty. In case $\mathcal{E}_k$ is empty, i.e., no cost comparison takes place at time t_k, then (25) is solved to generate the CAVs control input (lines 26–27).

Algorithm 2 Heuristic (H)

```
Input: $\mathbf{v}_k^{\mathrm{ref}}, \boldsymbol{x}_0, \mathbf{p}^{\mathrm{h}}, \mathbf{r}, \hat{\boldsymbol{\rho}}$
Output: $\mathcal{O}_k, \mathbf{w}$
 1: Obtain $\boldsymbol{x}_0, \mathbf{p}^{\mathrm{h}}$                 ▷ Initial states and prediction
 2: Solve one-time MIQP (23) $\Phi^{\mathrm{SMIQP}}$
 3: Obtain $\hat{\boldsymbol{\rho}}$ and $\mathcal{O}_0$ from $\mathbf{r}$
 4: Initialize $\mathcal{N}_0^{\mathrm{T}}$                 ▷ For CAVs behind mixed-platoons
 5: for $k \in \mathbb{R}^+$ do
 6:     Obtain $\boldsymbol{x}_0, \mathbf{p}^{\mathrm{h}}$             ▷ Current states and prediction
 7:     if $p_{i,k} \leq p^{\mathrm{in}},\ \forall i \in \mathcal{M}, \mathcal{N}$ then
 8:         for $i \in \mathcal{N}_k^{\mathrm{T}}$ do
 9:             Retrieve $\mathbf{w}$ from $k-1$ and compute $h_i^{\mathrm{s}}$
10:             if $\forall k : \exists\ h_i^{\mathrm{s}} > 0$ then     ▷ Any cons. violation
11:                 $s_i = s_i + 1$                 ▷ Violation counter
12:             end if
13:             if $s_i == n^{\max}$ then         ▷ Consistency check
14:                 Set $\mathcal{E}_k \leftarrow i$         ▷ Set of violating vehicles
15:             end if
16:         end for
17:         for $i \in \mathcal{E}_k$ do
18:             Obtain the alternative order
19:             Solve and compare $\hat{\Phi}_{i|j}^{\mathrm{FOQP}}$ and $\hat{\Phi}_{j|i}^{\mathrm{FOQP}}$
20:             Update $\mathcal{O}_{k+1}$ based on the cost
21:         end for
22:         if $\mathcal{O}_{k+1} \neq \mathcal{O}_k$ then           ▷ Any change of order
23:             Update $\mathcal{N}_{k+1}^{\mathrm{T}}$
24:         end if
25:     else
26:         Set $\mathcal{O}_k = \mathcal{O}_{k-1}$ and obtain $\mathbf{r}$
27:         Solve FOQP (25)
28:     end if
29:     Approximate $\hat{\boldsymbol{\rho}}$
30:     Apply $\mathbf{w}$ to CAVs of platoons
31: end for
```

VI. Numerical Simulations and Evaluation

In this section, we perform numerical simulations to evaluate the performance of our proposed heuristic (H) and compare it to the performance of the OMIQP (20) and the SMIQP (23) along with other heuristics, i.e., time-to-intersection (TTI) and the FCFS methods. The last two methods are explained in Section VI-A.

In all simulations, we consider a static number of vehicles approaching a single symmetric four-junction intersection for the sake of simplicity. Two main different scenarios are compared: the first one is the *nominal* case, where all leading vehicles are CAVs, schematized in Fig. 5; the second one is the *low disturbance* case, where one of the leading vehicles is an HDV. Furthermore, a third configuration with *high disturbance*, i.e., $N < M$ is briefly discussed in the end.

In the nominal case, all HDV trajectories can be partially regulated by the leading CAVs, which are able to slow them if necessary. In the disturbance cases, the platoons coordination is subject to noise stemming from uncertain human behavior, as the CAVs do not have any influence over the leading HDV. This allows us to evaluate the performance of our heuristic in realistic and non-ideal situations.

For each of the two cases above, we perform two different simulations. In the first one, we compare the heuristics with the MIQPs in terms of evaluation metrics, e.g., closed-loop cost, computation times, etc., as discussed in Section VI-B, executed once due to their high computational burden. In the second one, we simulate the heuristics ten times with different HDV input bounds to evaluate how consistent their performance can be. All simulations are subject to HDVs with additive normally distributed input noise Δu_k with average $\mu = 0$ and standard deviation $\sigma = 0.1$. The sampling time $\Delta t = 0.1$ s, while the duration of the simulations is $T^{\mathrm{sim}} = 8\ \ \mathrm{s}/K^{\mathrm{sim}} = 80$ steps and 10 s (100 steps) for the nominal and disturbance cases, respectively. The simulation takes longer for the latter because of the presence of the additional leading HDV.

Furthermore, we set the constraint relaxation weights $q^{e,q} = 1, q^{e,l} = 10^3$ when the order is fixed. To increase coordination safety, we introduce enlarged positions of the intersection within the vicinity of the CZ, where the lateral collision avoidance constraints (19) are active, i.e., the constraint is applied between $p^{\text{in}} - \delta^{\text{in}}$ and $p^{\text{out}} + \delta^{\text{out}}$. The values of δ^{in}, δ^{out} and all remaining constants are given in Table I.

All simulations are carried out using MATLAB with the CasADi framework [38] on a laptop with an Intel Core-i5 processor and 16 GB of RAM. BONMIN [39] is used to solve the MIQPs, while the continuous-relaxed QP problem and the fixed-order parametric versions (Subproblems A/B) in our heuristic are solved by IPOPT [40].

In these experiments, we utilize the fact that the HDV model prediction (4) is not exact [25]. To account for this notion, we define a different simulation model that switches between car-following and reference velocity tracking as follows:

$$u_{m,k} = \begin{cases} u_a, & \Delta p_{m,k} \geq \bar{d} \quad \forall m \in \mathcal{M} \\ u_b, & \Delta p_{m,k} < \bar{d} \end{cases} \tag{26}$$

with

$$u_a = k^v \left(v_{m,k} - v_{m,k}^{\text{ref}} \right) + \Delta u_k \tag{27a}$$

$$u_b = k^p \left(\Delta p_{m,k} - d^{\text{ref}} \right) + k^d \left(v_{m,k} - v_{i,k} \right) + \Delta u_k \tag{27b}$$

where k^v, k^p, k^d are weighting gains and we define $\Delta p_{m,k} := p_{i,k} - p_{m,k}$. Finally, $p_{i,k}$, $v_{i,k}$ are the position and velocity of the vehicle immediately in front of the HDV m, and Δu_k is an additive noise that represents uncertainty in the human driver's behavior.

A. Alternative Heuristics

Let us introduce the next two alternative simple heuristics that have been proposed in the literature: the FCFS heuristic as described in [24] and the TTI heuristic [30]. Despite the difference in the names, the two approaches are very similar to each other. FCFS is obtained by sorting the platoon's priority based on the order of entering the intersection area, which translates to $\mathcal{O}_k$. Afterward, the crossing order remains fixed until all vehicles clear the intersection, i.e., its current order $\mathcal{O}_k = \mathcal{O}_{k-1}$ is kept the same as before at time $k-1$.

TTI sorts the platoons by their estimated time to reach the intersection entry, defined as follows:

$$t_{i,k}^{\text{TTI}} = \frac{p^{\text{in}} - p_{i,k}}{v_{i,k}}. \tag{28}$$

The platoon with the shortest time is put first in $\mathcal{O}_k$. This step is executed in lines 4–5 of Algorithm 3. Similar to what we do in our heuristic, TTI updates the crossing order until the front vehicle becomes too close to the intersection, see line 7. The main difference with respect to FCFS is that, while TTI updates the crossing order throughout the simulation, FCFS keeps it fixed. For both FCFS and TTI, at time $k = 0$, the binaries $\mathbf{r}$ are computed by solving (25) without enforcing safety constraint (12). For $k > 0$, variables $\mathbf{r}$ are obtained from the previously computed trajectory, as we do in our heuristic (see line 9 of Algorithm 3). The resulting $\mathcal{O}_k$, $\mathbf{r}$ are then passed to (25) to get an optimal acceleration/deceleration profile (line 10). Both FCFS and TTI are relatively simple and, hence, computationally fast.

Algorithm 3 Time-to-Intersection

```
Input: $\mathbf{v}_k^{\text{ref}}, \boldsymbol{x}_0, \mathbf{p}^{\text{h}}, \mathbf{r}, \hat{\boldsymbol{\rho}}$
Output: $\mathcal{O}_k$, $\mathbf{w}$
1:  for $k \in \mathbb{R}^+$ do
2:      Obtain $\mathbf{v}_k^{\text{ref}}, \boldsymbol{x}_0, \mathbf{p}^{\text{h}}$        ▷ Parameter
3:      if $p_{i,k} \leq p^{\text{in}}$, $\forall j \in \mathcal{M}, \mathcal{N}$ then
4:          Compute $t_{i,k}^{\text{TTI}}$
5:          Sort $t_{i,k}^{\text{TTI}}$ to obtain $\mathcal{O}_k$ and $\mathbf{r}$
6:      else
7:          Set $\mathcal{O}_k = \mathcal{O}_{k-1}$ and obtain $\mathbf{r}$
8:      end if
9:      Approximate $\hat{\boldsymbol{\rho}}$ from previous trajectory
10:     Solve fixed-order QP (25)
11: end for
```

B. Evaluation Metrics

The following metrics are introduced to evaluate the closed-loop performance of the aforementioned methods, which are presented in Figs. 6–11 and Tables II–V.

1) *Crossing order:* We monitor the crossing order $\mathcal{O}_k$ and its evolution over the simulation time T^{sim}.

2) *Cardinality and timing of reordering:* We record the times at which the crossing order is changed (reordering/switching), i.e., $\mathcal{O}_k \neq \mathcal{O}_{k-1}$ are recorded in vector $\boldsymbol{\tau}^{\text{OMIQP}} = [\tau_1^{\text{OMIQP}}, \tau_2^{\text{OMIQP}}, \ldots]$, $\boldsymbol{\tau}^{\text{SMIQP}}$, $\boldsymbol{\tau}^H$ for the OMIQP, SMIQP, and heuristic, respectively. Their respective cardinality is expressed as $|\boldsymbol{\tau}|$.

3) *Closed-loop cost:* We record the closed-loop total objective function (cost) values from all N CAVs over the simulation duration T^{sim} s or in K^{sim} steps

$$\Phi^{\text{cl}} := \Phi^{\text{cl,SI}} + \sum_{k=0}^{K^{\text{sim}}} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} J^e \left(\eta_{i,j,k} \right) \tag{29}$$

$$\Phi^{\text{cl,SI}} := \sum_{k=0}^{K^{\text{sim}}} \sum_{i=1}^{N} q^v \left(v_{i,k}^{\text{ref}} - v_{i,k} \right)^2 + q^u u_{i,k}^2. \tag{30}$$

Note that we also define the closed-loop cost without the slack terms $\Phi^{\text{cl,SI}}$ to evaluate the cost of the applied state and input reference tracking.

4) *Maximum constraint violation:* We record the worst case slack from each pair of platoons used to relax the constraint (19) within the safety margins $p^{\text{in}} - \delta^{\text{in}}$ and $p^{\text{out}} + \delta^{\text{out}}$ over T^{sim}

$$\boldsymbol{\eta}^{\max} := \max_{k,j,i} \; \eta_{i,j,k} \tag{31a}$$

$$\text{s.t. } k \in \left\{ 0, \ldots, K^{\text{sim}} \right\} \tag{31b}$$

$$i, j \in \{1, \ldots, N\}^2 \tag{31c}$$

$$i \neq j. \tag{31d}$$

Note that this only accounts for the *current* slack and not the future violation.

TABLE I
CONSTANT VALUES

Constant	Value	Constant	Value
T^{pre} / K^{pre}	2.6 or 3.5 s / 26 or 35	d^{ref}	9 m
q^{v}	10	$\bar{l}$	2 m
d^{min}	4 m	v^{nom}	40 km/h
q^{u}	1	n^{max}	3
$q^{\text{e,l}}$	10 or 1000	k^{v}	1
M^{b}	1000	k^{p}	2
Δt	0.1 s	k^{d}	1
T^{sim} / K^{sim}	8 or 10 s / 80 or 100	$\bar{d}$	7 m
$v^{\text{min/max}}$	$3.6 - 70$ km/h	$p^{\text{in/out}}$	± 2
$u^{\text{min/max}}$	± 3 m/s^2	δ^{in}	13 m
$q^{\text{e,q}}$	1	δ^{out}	8 m

5) *RMS of acceleration:* We record the root mean square (rms) of the acceleration and deceleration of the CAVs, i.e., control input. This metric shows the average amount of control actions injected into each CAV at each time t_k

$$\mathbf{u}^{\text{rms}} := \sqrt{\frac{1}{N \times K^{\text{sim}}} \sum_{i=1}^{N} \sum_{k=0}^{K^{\text{sim}}} u_{i,k}^2}. \tag{32}$$

6) *Computation Time:* We record the worst case computation time, over T^{sim} in seconds (s) required for solving the crossing order problem at a single time step t_k, i.e.,

$$\mathbf{t}^{\max} := \max_{t_k} \mathbf{t}^{\text{comp}} \tag{33}$$

where $\mathbf{t}^{\text{comp}}$ is a vector collecting the computation times from a single simulation.

Remark 1: Note that for the sake of simplicity, we chose a scenario in which q^v and q^u are the same for all vehicles. Consequently, FCFS and TTI can be expected to perform well. In case these cost parameters were different among the vehicles, we would observe a much better performance of our heuristic, as it explicitly accounts for this, while FCFS and TTI cannot. As we will detail below, even in this unfavorable comparison setting, our heuristic performs well compared to FCFS and TTI.

C. Nominal Scenario

We consider now the case in which all the leading vehicles are CAVs, as shown in Fig. 5, where the three CAVs are colored blue and the two HDVs are colored red. Platoon 2 is composed of CAV 2 and HDV 4; platoon 3 is composed of CAV 3 and HDV 5; and CAV 1 is a one-vehicle platoon.

The first vehicle on the line is CAV 2, which starts at a distance of 53 m from the center of CZ. The rest of the vehicles in the line, with the sequence of HDV 4 - CAV 3 - HDV 5 - CAV 1, are placed with a gap of 7.5 m between each other. The initial speed of all vehicles is $v_i^0 = 50$ km/h (13.89 m/s), and the nominal reference speed is $v^{\text{nom}} = 60$ km/h (16.67 m/s). In order to force a switch in the crossing order, HDV 4 is set to slow down to 23 km/h (6.39 m/s). The reordering problem is solved in a closed loop until CAV 1 has reached the CZ, and the order is fixed after that time.

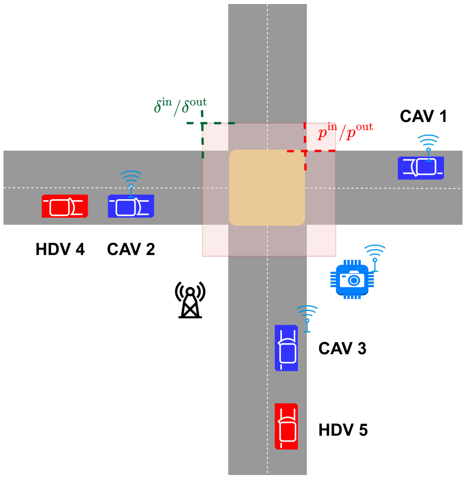
$\delta^{\text{in}}/\delta^{\text{out}}$
$p^{\text{in}}/p^{\text{out}}$
CAV 1
HDV 4
CAV 2
CAV 3
HDV 5

Fig. 5. Vehicle configuration for *nominal* case.

1) Simulation Against MIQPs: We first compare the two MIQP formulations, i.e., OMIQP (20) and SMIQP (23). To solve the MIQPs within a reasonable time, we limit their iterations to 1.7×10^4 and perform only a single simulation. To make a fair comparison between the MIQPs and the heuristics, the consistency check is not used here, i.e., $n^{\max} = 0$.

Fig. 6 shows the crossing order [(a) and (b)] and cost [(c) and (d)] generated by the simulation. It can be observed that all vehicles begin to move in the crossing order dictated by their initial positions, i.e., $\mathcal{O}_{k=0} = [o_{1,0}, o_{2,0}, o_{3,0}]^\top = [2, 3, 1]^\top$. In Fig. 6(a), it can be observed that both MIQPs yield crossing orders that chatter between different orders until the order finally settles. This issue is the consequence of the additive perturbation, which, in case two orders have very similar costs, makes the MIQP solvers alternate between one order and the other.

As one can see in Fig. 7(c) and (d), HDV 4 is set to slow down, such that platoons 2 and 3 also have to slow down, until the order is changed by swapping platoons 2 and 3. SMIQP eventually settles at $\mathcal{O}_k = [3, 2, 1]^\top$ at time $\tau_6^{\text{SMIQP}} = 1.3$ s [Fig. 6(a)]. Our heuristic follows the same reordering at $\tau_1^H = 1.5$ s [Fig. 6(b)] without any chattering. The difference in the switching time between the methods is caused by the approximate nature of the timing binaries $\boldsymbol{\rho}$ used by our heuristic. When solving the FOQPs (25) relative to the current and alternative orders in our heuristic, $\hat{\boldsymbol{\rho}}$ are parameters approximated from the predicted trajectories at the previous time t_k, while in SMIQP they are decision variables. OMIQP eventually also yields the same order at a later time $\tau_4^{\text{OMIQP}} = 2.0$ s. This is due to the higher complexity of the OMIQP formulation, which entails that the optimal solution cannot be obtained within the imposed iteration limit.

The same pattern repeats also for the next switching time, which occurs at time $\tau_7^{\text{SMIQP}} = 1.5$ s for SMIQP and at time $\tau_2^H = 1.9$ s for our heuristic. They converge to the final order $\mathcal{O}_k = [3, 1, 2]^\top$. The final order is also obtained by OMIQP at a later time $\tau_5^{\text{OMIQP}} = 4.4$ s. The motivation for the

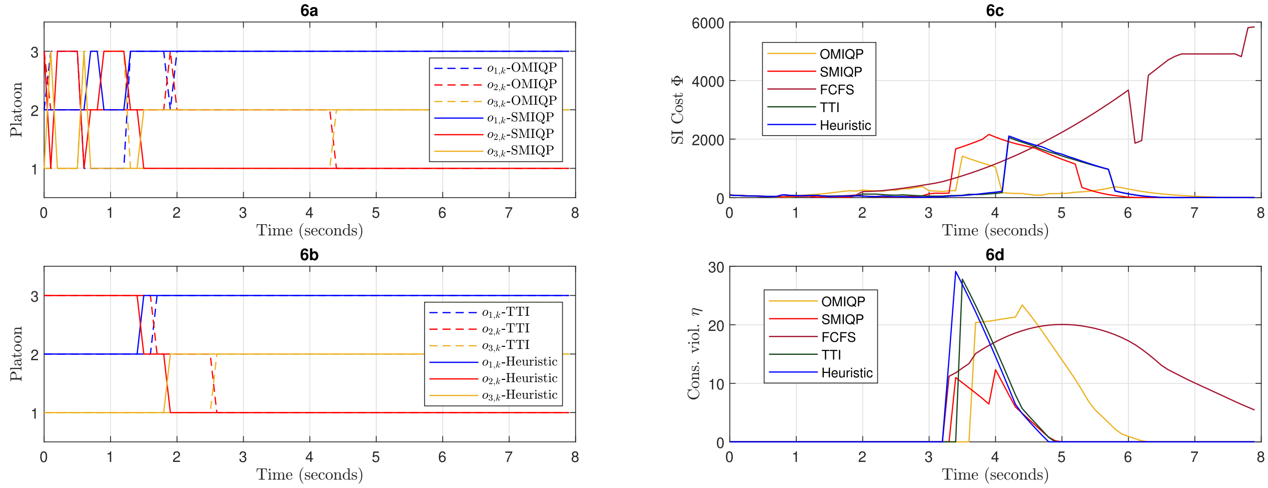

Fig. 6. Crossing order in subfigures (a) and (b), closed-loop cost evolution, and constraint violation in subfigures (c) and (d) from the nominal test for all methods. Subfigure (a) displays the crossing order of the benchmark simplified MIQP (SMIQP) (dashed lines), and the original MIQP (OMIQP) (solid lines), while subfigure (b) displays the crossing order from TTI and our heuristic (solid lines). The line colors represent the index inside $\mathcal{O}_k$. FCFS is omitted here to maintain readability but its order remains the same all the time, i.e., $\mathcal{O}_k = [2, 3, 1]^\top$. Similarly, subfigure (c) displays the closed-loop SI costs from all methods, while the subfigure (d) displays the constraint violation.

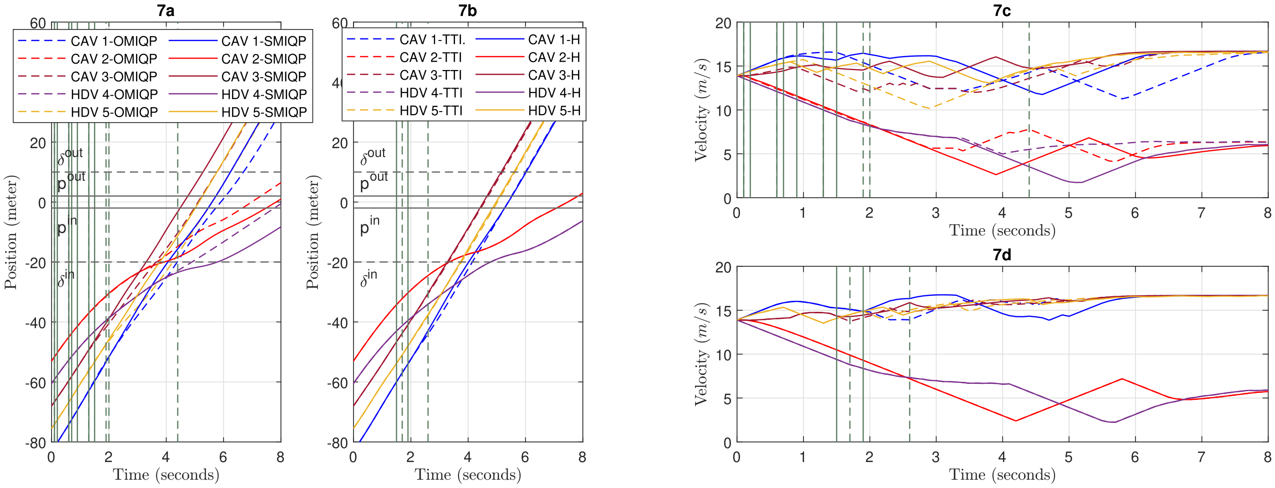

Fig. 7. Position in subfigures (a) and (b) and velocity in subfigures (c) and (d) from the *nominal* test for all methods. Subfigure (a) is relative to the benchmark SMIQP (solid lines), and OMIQP (dashed lines), while subfigure (b) is relative to TTI and heuristic. subfigure (c) displays the velocity of the MIQPs, while subfigure (d) displays the velocity from TTI and our heuristic. The legends are applied to all subfigures. FCFS is omitted here to maintain readability. The vertical green dashed and solid lines in each subfigure indicate reordering timings τ of OMIQP/TTI, and SMIQP/heuristic, respectively.

TABLE II
PERFORMANCE COMPARISON FOR NOMINAL SCENARIO

Methods	$\lvert\boldsymbol{\tau}\rvert$	Total cost Φ^{cl}	SI cost $\Phi^{\text{cl,SI}}$	$\boldsymbol{\eta}^{\max}$ [m]	$\mathbf{u}^{\text{RMS}}$ [m/s^2]	$\mathbf{t}^{\max}$ [s]	Times faster than OMIQP
OMIQP	7	133812	18531	13.21	2.37	234.03	N/A
SMIQP	6	47279	35545	9.27	2.16	156.50	1.49
FCFS	0	633800	146809	14.38	2.13	0.16	1464.92
TTI	2	62037	27787	17.05	1.97	0.06	3440.45
Heuristic	2	73117	27853	15.85	1.95	1.24	189.01

reordering decisions is illustrated in Fig. 8, which presents the cost-to-go comparison run inside our heuristic between the QPs of current (first) and alternative orders. If the current order were kept, the cost would continue to increase, so that a reordering yields a lower cost. TTI updates the crossing order similar to our heuristic, with two reordering events having the same pair-swaps, though at a slightly later time compared to our heuristic. No chattering is therefore present, as shown in Fig. 6(b). Consequently, all the evaluation metrics yield similar values for TTI and our heuristic.

In Table II, one can see that the sum of the closed-loop state reference tracking and input (SI) cost $\Phi^{\text{cl,SI}}$ of our heuristic is higher than the one of OMIQP, but lower than the one of SMIQP. SMIQP yields the highest $\Phi^{\text{cl,SI}}$ here because it needs to minimize future constraint violations. Consequently, it triggers reordering early, which requires larger inputs and

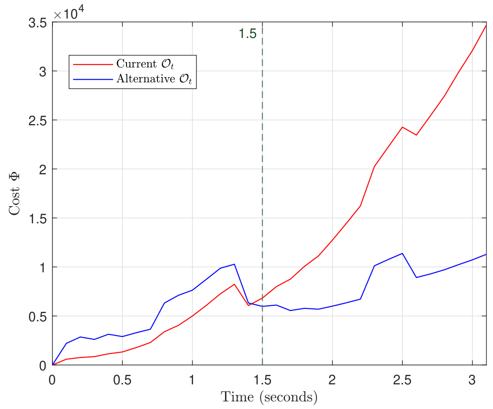

Fig. 8. Cost-to-go evolution of the current and alternative QPs solved inside our heuristic.

deviations from the reference speed. One can see that CAV 2 significantly decelerates, as shown in Fig. 7. In Fig. 6(c), we see that this results in the $\Phi^{\mathrm{cl,SI}}$ of SMIQP being the highest around $t = 3.4 - 4$ s. On the other hand, it yields the lowest maximum constraint violation $\boldsymbol{\eta}^{\max}$, which, as depicted in the left subfigure of Fig. 7, occurs around the entrance safety margin δ^{in} and before CZ hence practically safe. The early reorderings also imply that the SMIQP Φ^{cl} is the lowest.

In Fig. 6(c), it is seen that our heuristic and TTI yield an evolution of $\Phi^{\mathrm{cl,SI}}$ similar to SMIQP, but slightly delayed. This yields lower $\Phi^{\mathrm{cl,SI}}$ w.r.t. SMIQP, which is due to the lower use of inputs and reference deviation for swapping at later times, but consequently, $\boldsymbol{\eta}^{\max}$ and Φ^{cl} are higher than those of SMIQP.

Furthermore, we can see that OMIQP has the lowest $\Phi^{\mathrm{cl,SI}}$. This is due to the fact that the platoons are trying to swap orders even though reordering decisions come late because the solver does not reach convergence. Hence, it can be seen in Fig. 6(c), that in around $t = 1 - 3$ s, OMIQP has the highest $\Phi^{\mathrm{cl,SI}}$, but the required inputs and reference deviation for swapping later are eventually the lowest. Furthermore, this decision results in more use of slacks that significantly contribute to the Φ^{cl} resulting in the OMIQP having the highest Φ^{cl}, as in Fig. 6(d). Due to its fixed order policy, we can see in Fig. 6(c) that FCFS yields a high cost.

With regard to the average (rms) control input $\mathbf{u}^{\mathrm{rms}}$ exhibited by each CAV, it can be observed that the gap between the SMIQP and our heuristic or TTI is much smaller than the gap in the cost. Due to the imposed iteration bound, as expected, OMIQP yields the largest $\mathbf{u}^{\mathrm{rms}}$, worse than FCFS.

Fig. 7 presents the positions and velocities obtained by MIQPs and our heuristic. One can observe from the velocity curves in Fig. 7(c) and (d) that SMIQP and our heuristic force platoon 3 to only marginally slow down until reordering is executed. It is noteworthy to observe that SMIQP and our heuristic yield very similar position and velocity profiles. Due to later reordering times, OMIQP forces CAV 3 and 1 to slow down for a longer time and, consequently, aside from platoon 2, all platoons enter the CZ at a later time, compared to SMIQP and our heuristic. As $\mathcal{O}_k$ yielded by our heuristic and TTI are very close, their position and velocity profiles are almost the same.

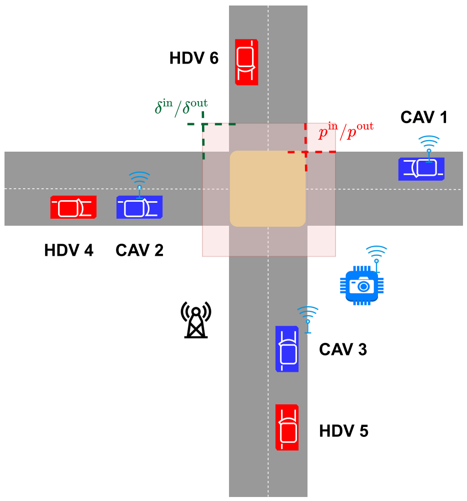
HDV 6
$\delta^{\mathrm{in}}/\delta^{\mathrm{out}}$
$p^{\mathrm{in}}/p^{\mathrm{out}}$
CAV 1
HDV 4
CAV 2
CAV 3
HDV 5

Fig. 9. Vehicle configuration for *disturbance* case.

The main advantage of our heuristic over MIQPs is shown in the (worst case) computation time $\mathbf{t}^{\max}$ reported in Table II: our heuristic is about 189 times faster to solve than OMIQP and also about a hundred times faster than SMIQP. SMIQP is about 1.5 times faster than OMIQP, but that does not make any significant difference in the context of time-tractable coordination. FCFS and TTI are up to a thousand times faster than OMIQP thanks to their simplicity.

2) Simulation With Random HDV Input Bounds: In this second set of simulations, random HDV input bounds are sampled from a uniform distribution within 10% of the nominal ones (± 3 m/s^2). The consistency check is used with $n^{\max} = 3$, as detailed in Table I. As mentioned before, MIQPs are not executed here due to their long computation times, which means only FCFS, TTI, and our heuristic are considered.

We provide in Table III the values of the metrics averaged over ten simulations. One can see that even though the input bounds are different in each simulation, our heuristic, and TTI perform reordering only twice. The consistency check plays an important role here, as it makes sure that an order swap only occurs when it is advantageous enough, therefore mitigating the chattering behavior of the MIQPs as observed in Section VI-C1. The values of $\boldsymbol{\eta}^{\max}$, Φ^{cl}, and $\Phi^{\mathrm{cl,SI}}$ are slightly higher for our heuristic than in the previous simulation in Section VI-C1, but the average acceleration remains almost the same. In terms of $\mathbf{t}^{\max}$, one can see that the FCFS, TTI, and our heuristic are consistently faster compared to the MIQPs in the previous simulations. Our heuristic has a lower Φ^{cl} and approximately the same $\Phi^{\mathrm{cl,SI}}$ compared to TTI. FCFS is the worst method in all aspects except $\boldsymbol{\eta}^{\max}$ and $\mathbf{t}^{\max}$.

D. Low Disturbance Scenario

This scenario includes an additional *leading* HDV 6 alongside existing vehicles, as shown in Fig. 9, acting as a *disturbance* to the platoons' coordination. Without any CAV in front that can regulate HDV 6, all CAVs are imposed an additional safety constraint by the IM: HDV 6 must occupy the intersection first. HDV 6 tracks a constant speed $v_i^{\mathrm{ref}} =$

TABLE III
AVERAGE PERFORMANCE FOR NOMINAL SCENARIO WITH DIFFERENT INPUT BOUNDS

Methods	$\lvert\boldsymbol{\tau}\rvert$	**Average** τ	**Total cost** Φ^{cl}	**SI cost** $\Phi^{\mathrm{cl,SI}}$	$\boldsymbol{\eta}^{\max}$ **[m]**	$\mathbf{u}^{\mathrm{RMS}}$ **[m/s^2]**	$\mathbf{t}^{\max}$ **[s]**
FCFS	0	N/A	417662	149077	11.09	2.11	0.10
TTI	2	2.33	247608	30017	13.66	1.90	0.07
Heuristic	2	1.96	106286	30187	12.25	1.97	2.42

TABLE IV
PERFORMANCE COMPARISON FOR DISTURBANCE SCENARIO

Methods	$\lvert\boldsymbol{\tau}\rvert$	**Total cost** Φ^{cl}	**SI cost** $\Phi^{\mathrm{cl,SI}}$	$\boldsymbol{\eta}^{\max}$ **[m]**	$\mathbf{u}^{\mathrm{RMS}}$ **[m/s^2]**	$\mathbf{t}^{\max}$ **[s]**	**Times faster than SMIQP**
SMIQP	14	160883.32	106109.15	19.83	2.42	752.64	N/A
FCFS	0	202020.83	131094.95	8.7	2.48	0.083	6770.35
TTI	2	407715.49	41222.23	19.73	2.34	0.085	6598.52
Heuristic	2	157617.62	84268.94	12.43	2.37	1.97	281.17

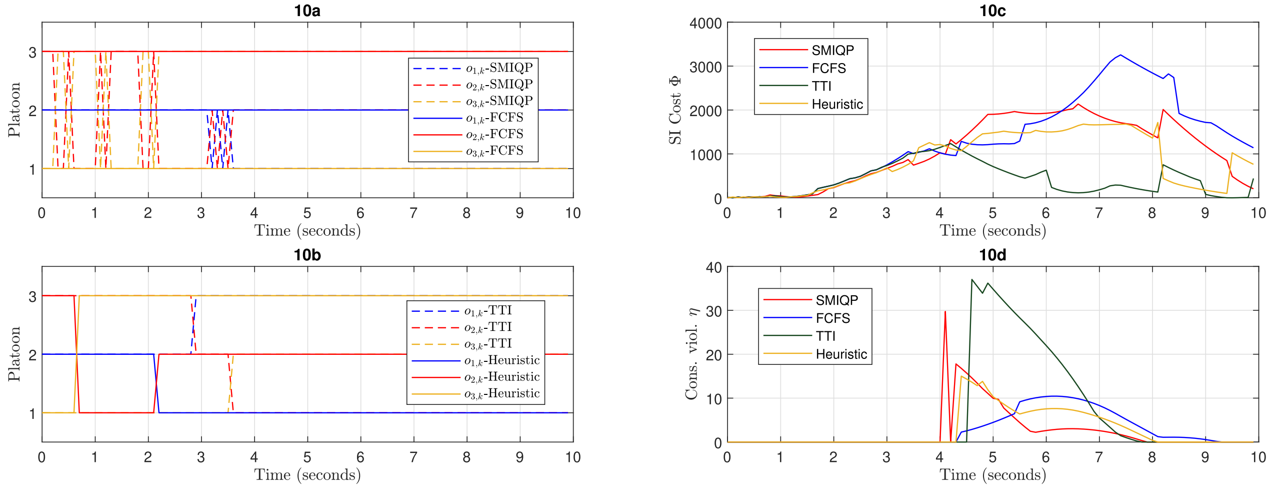

Fig. 10. Crossing order in subfigures (a) and (b) and closed-loop SI cost evolution in subfigures (c) and (d) from the *disturbance* test for all methods. Subfigure (a) displays the crossing order of SMIQP (dashed lines) and FCFS (solid lines), while subfigure (b) is relative to TTI (dashed lines) and our heuristic (solid lines). Subfigure (c) displays the costs while subfigure (d) displays the constraint violations for all methods.

60 km/h and is initially positioned at $86,25$ m from the center of the CZ with an initial velocity of 50 km/h. It is also modeled using the switching model (4). The initial position of CAV 2 is shifted to 60 m from the center of the CZ, and the rest of the vehicles are configured with the same sequence from CAV 2 as in the previous nominal case, also with 7.5 m gap each. Here, HDV 4 does not slow down and instead follows v^{nom}. The prediction and simulation horizon are selected as $T^{\mathrm{pre}} = 3.5$ s and $T^{\mathrm{sim}} = 10$ s. Finally, we select $\bar{d} = 8$ m and $d^{\mathrm{ref}} = 10$ m.

In these simulations, we drop OMIQP as the scenario is more complex than before, and the solver does not manage to converge in a reasonable time. For SMIQP, the imposed iteration limit is 10^4.

1) Simulation Against SMIQP: As shown in Figs. 10(a) and (b) and 11(a) and (b), the initial order is dictated by the initial positions of the vehicles, that is, $\mathcal{O}_0 = [2, 3, 1]^\top$. Due to the additional safety constraint against HDV 6, this causes the platoons to gradually decelerate, except for platoon (CAV) 1. In the beginning, platoon 2 has the highest deceleration as it is the closest to the intersection and needs to slow down to let HDV 6 cross first.

As platoons 2 and 3 slow down, the tail HDVs behind them are also forced to decelerate. This action is necessary to prevent the possibility that HDVs 4 and 5 occupy the CZ at the same time as HDV 6, which can result in a potential collision. Platoon 1 (which is a one-vehicle platoon) is then given the opportunity to overtake both of them gradually, as can be seen in Fig. 11.

In Fig. 10(a), it can be observed that SMIQP chatters between $\mathcal{O}_k = [2, 3, 1]^\top$ and $\mathcal{O}_k = [2, 1, 3]^\top$ starting at $\tau_1^{\mathrm{SMIQP}} = 0.3$ s. The order eventually settles to the latter at $\tau_9^{\mathrm{SMIQP}} = 2.2$ s. Instead, our heuristic requires only a single-order swap at $\tau_1^{H.} = 0.7$ s. Finally, the crossing order is further modified to $\mathcal{O}_k = [1, 2, 3]^\top$ by both approaches. In this case, SMIQP switches last, with some chattering as well, which is due to the imposed iteration limit as observed in the solver outputs. As in Section VI-C, TTI and our heuristic can avoid chattering without consistency check. TTI has two reorderings at $\tau_1^{\mathrm{TTI}} = 2.8$ s and $\tau_2^{\mathrm{TTI}} = 3.5$ s. However, their final order is different as TTI converges to $\mathcal{O}_k = [3, 1, 2]^\top$.

In Table IV, $\Phi^{\mathrm{cl,SI}}$ $\Phi^{\mathrm{cl,SI}}$ yielded by our heuristic are lower than those of SMIQP, but the opposite is true for Φ^{cl}. This indicates the same trade-off observed in the previous scenario,

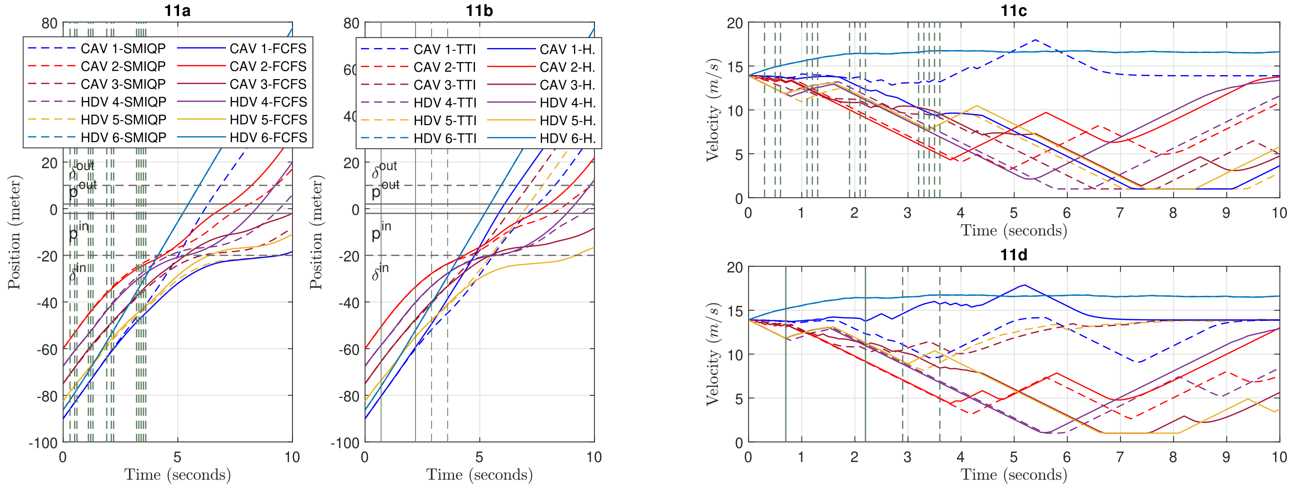

Fig. 11. Position in subfigures (a) and (b) and velocity in subfigures (c) and (d) from the *disturbance* test for all methods. Subfigure (a) is relative to SMIQP and FCFS, while subfigure (b) is relative to TTI and our heuristic. Subfigure (c) displays the velocity of the benchmark, while subfigure (d) displays the velocity of our heuristic. The color and lines from the legends are applied to all subfigures. The solid and dashed-dotted vertical green lines indicate reordering timings τ that are relative to SMIQP/TTI and heuristic, respectively.

TABLE V

Average Performance Comparison for Disturbance Scenario With Different Input Bounds

Methods	$\lvert\boldsymbol{\tau}\rvert$	**Average** $\boldsymbol{\tau}$	**Total cost** Φ^{cl}	**SI cost** $\Phi^{\mathrm{cl,SI}}$	$\boldsymbol{\eta}^{\max}$ **[m]**	$\mathbf{u}^{\mathrm{RMS}}$ **[m/s^2]**	$\mathbf{t}^{\max}$ **[s]**
FCFS	0	N/A	193124.29	146061.89	13.32	2.45	0.28
TTI	2	3.20	257279.56	55646.11	21.68	2.11	0.31
Heuristic	2	1.32	142941.34	78513.10	15.50	2.39	1.89

where the early reordering performed by SMIQP yields lower constraint violation at the price of slightly higher $\Phi^{\mathrm{cl,SI}}$ and $\mathbf{u}^{\mathrm{rms}}$. This is displayed in Fig. 10 where the cost evolution is initially similar for all approaches, but starts to differ around $t = 4.5$ s where, due to the early reordering, the cost is slightly higher for SMIQP. Due to the subsequent reorderings, our heuristic has a slightly higher constraint violation from $t = 5$ s. Additionally, one can see that the worst case $\boldsymbol{\eta}^{\max}$ of SMIQP is slightly higher than that of our heuristic, due to the iteration limit which hinders convergence for SMIQP. TTI has the lowest $\Phi^{\mathrm{cl,SI}}$, at the expense of the highest Φ^{cl} and $\boldsymbol{\eta}^{\max}$, due to a different final crossing order which causes higher violation. Finally, FCFS has the highest $\Phi^{\mathrm{cl,SI}}$ and $\mathbf{u}^{\mathrm{rms}}$ due to its fixed order, although it has the lowest $\boldsymbol{\eta}^{\max}$.

As seen in Section VI-C1, the approximation of timing binaries in our heuristic leads to differences in reordering. However, our heuristic is shown to yield significantly smaller $\mathbf{t}^{\max}$, making it about 280 times faster than SMIQP, with a better consistency of the crossing order and close to optimal trajectories. FCFS and TTI remain the fastest methods here, even though their Φ^{cl} or $\boldsymbol{\eta}^{\max}$ can be far from optimal.

2) Simulation With Random HDV Inputs: In this comparison, we further conduct ten different simulations with randomized HDV input bounds, with a similar setting as in the previous simulation in Section VI-D1. The input bounds are generated analogously to those of Section VI-C2. The consistency check is applied here with $n^{\max} = 3$. SMIQP is not executed here due to its exceedingly long computation time.

The resulting values of the metrics averaged over the simulations are presented in Table V. One can see that both TTI and our heuristic consistently yield 2 reordering events in each simulation, such that chattering is avoided. Regarding $\boldsymbol{\eta}^{\max}$, one can see that our heuristic outperforms TTI and FCFS, even though TTI has the lowest $\Phi^{\mathrm{cl,SI}}$. FCFS has the worst $\Phi^{\mathrm{cl,SI}}$ but the lowest $\boldsymbol{\eta}^{\max}$. Finally, $\mathbf{t}^{\max}$ and $\mathbf{u}^{\mathrm{rms}}$ of all methods remain small. In general, the pattern is similar to that of the nominal case in Section VI-C2.

E. High Disturbance Scenario

As a last comparison, we test our algorithm in a scenario with a lower ratio of CAVs against HDVs. In addition to the vehicles in Fig. 9, two additional *leading* HDVs 7 and 8 are inserted in front of CAVs 1 and 3, respectively. The number of CAVs N remains the same to allow for direct comparison with the previous scenarios, while now $M = 5$. As before, due to the presence of leading HDVs, the CAVs have to satisfy additional safety restrictions. This increases the problem complexity, which rules out the use of MIQPs. Ten simulations are carried out with different input bounds and averaged as in Section VI-C2.

From the simulation with the three methods (FCFS, TTI, and our heuristic), we see that the higher presence of the HDVs makes coordinating the platoons more difficult, as the leading HDVs trajectories cannot be controlled. Consequently, the CAVs have to decelerate so that all leading HDVs occupy the intersection first. This increases the $\Phi^{\mathrm{cl,SI}}$ of TTI and our heuristic, which are averaged over the ten simulations, 135 140.92 and 105 608.29, respectively.

FCFS $\Phi^{\mathrm{cl,SI}}$ is $134\,251.47$, which is slightly smaller than TTI.

Both for TTI and our heuristic, we observe that reordering still occurs between the CAV-led platoons in an attempt to maintain feasibility and minimize the cost. The overall results here show that the algorithm performance may degrade with lower CAV penetrations. This is to be expected, due to the increased uncertainty and decreased control authority.

VII. Conclusion

We proposed a heuristic algorithm designed to coordinate both CAVs and HDVs at unsignalized intersections. Our algorithm is able to efficiently handle dynamic reordering problems that may arise due to changes in HDVs' trajectories. Obtaining the optimal crossing order and acceleration profiles requires solving computationally expensive MIQPs, which are not real-time feasible. Our heuristic overcomes this challenge by combining a problem-tailored future constraint violation/consistency check and a cost comparison strategy. The consistency check allows us to restrict reordering actions to a smaller subset of platoons, therefore mitigating subproblems branching complexity. The cost comparison eventually decides whether an order change takes place or not.

Simulation results from several scenarios demonstrate that our heuristic is orders of magnitude faster than solving MIQPs and has better order consistency, at the price of a marginal performance degradation. Also, our heuristic generally has better solutions compared to the FCFS and TTI results in unfavorable settings. Additionally, we see that the SI cost generally increases when the CAV penetration rate decreases.

Future work will further develop numerical aspects of our heuristic to improve its performance while addressing more challenging and practical scenarios, e.g., involving a continuous flow of vehicles simulated using traffic simulators or the use of learning-based methods.

References

[1] E.-H. Choi, "Crash factors in intersection-related crashes: An on-scene perspective," Amer. Psychol. Assoc., Washington, DC, USA, Tech. Rep. DOT HS 811 366, 2010.

[2] V. Silva, C. Siebra, and A. Subramanian, "Intersections management for autonomous vehicles: A heuristic approach," *J. Heuristics*, vol. 28, no. 1, pp. 1–21, Feb. 2022.

[3] G. Sharon and P. Stone, "A protocol for mixed autonomous and human-operated vehicles at intersections," in *Proc. Int. Conf. Auto. Agents Multiagent Syst. (AAMAS)*, in Lecture Notes in Computer Science, São Paulo, Brazil, May 2017, pp. 151–167.

[4] A. I. M. Medina, F. Creemers, E. Lefeber, and N. van de Wouw, "Optimal access management for cooperative intersection control," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 5, pp. 2114–2127, May 2020.

[5] R. Hult, M. Zanon, S. Gros, and P. Falcone, "Primal decomposition of the optimal coordination of vehicles at traffic intersections," in *Proc. IEEE 55th Conf. Decis. Control (CDC)*, Dec. 2016, pp. 2567–2573.

[6] M. B. Mertens, J. Müller, and M. Buchholz, "Cooperative maneuver planning for mixed traffic at unsignalized intersections using probabilistic predictions," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Aachen, Germany, Jun. 2022, pp. 1174–1180.

[7] A. P. Chouhan and G. Banda, "Autonomous intersection management: A heuristic approach," *IEEE Access*, vol. 6, pp. 53287–53295, 2018.

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

[9] L. Makarem and D. Gillet, "Model predictive coordination of autonomous vehicles crossing intersections," in *Proc. 16th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2013, pp. 1799–1804.

[10] 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. Transp. Syst. (ITSC)*, Oct. 2013, pp. 1456–1461.

[11] G. R. de Campos, P. Falcone, R. Hult, H. Wymeersch, and J. Sjöberg, "Traffic coordination at road intersections: Autonomous decision-making algorithms using model-based heuristics," *IEEE Intell. Transp. Syst. Mag.*, vol. 9, no. 1, pp. 8–21, Spring 2017.

[12] R. Hult, M. Zanon, S. Gras, and P. Falcone, "An MIQP-based heuristic for optimal coordination of vehicles at intersections," in *Proc. IEEE Conf. Decis. Control (CDC)*, Dec. 2018, pp. 2783–2790.

[13] N. Murgovski, G. R. de Campos, and J. Sjöberg, "Convex modeling of conflict resolution at traffic intersections," in *Proc. 54th IEEE Conf. Decis. Control (CDC)*, Dec. 2015, pp. 4708–4713.

[14] J. Karlsson et al., "Intersection crossing with reduced number of conflicts," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2018, pp. 1993–1999.

[15] Z. Deng, K. Yang, W. Shen, and Y. Shi, "Cooperative platoon formation of connected and autonomous vehicles: Toward efficient merging coordination at unsignalized intersections," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 5, pp. 5625–5639, May 2023.

[16] 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.

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

[18] A. M. I. Mahbub, A. A. Malikopoulos, and L. Zhao, "Decentralized optimal coordination of connected and automated vehicles for multiple traffic scenarios," *Automatica*, vol. 117, Jul. 2020, Art. no. 108958.

[19] B. Chalaki and A. A. Malikopoulos, "An optimal coordination framework for connected and automated vehicles in two interconnected intersections," in *Proc. IEEE Conf. Control Technol. Appl. (CCTA)*, Hong Kong, Aug. 2019, pp. 888–893.

[20] S. Aoki and R. Rajkumar, "V2 V-based synchronous intersection protocols for mixed traffic of human-driven and self-driving vehicles," in *Proc. IEEE 25th Int. Conf. Embedded Real-Time Comput. Syst. Appl. (RTCSA)*, Aug. 2019, pp. 1–11.

[21] R. Chen, J. Hu, M. W. Levin, and D. Rey, "Stability-based analysis of autonomous intersection management with pedestrians," *Transp. Res. C, Emerg. Technol.*, vol. 114, pp. 463–483, May 2020.

[22] B. Peng, M. F. Keskin, B. Kulcsár, and H. Wymeersch, "Connected autonomous vehicles for improving mixed traffic efficiency in unsignalized intersections with deep reinforcement learning," *Commun. Transp. Res.*, vol. 1, Dec. 2021, Art. no. 100017.

[23] A. M. I. Mahbub, V.-A. Le, and A. A. Malikopoulos, "A safety-prioritized receding horizon control framework for platoon formation in a mixed traffic environment," *Automatica*, vol. 155, Sep. 2023, Art. no. 111115.

[24] Z. Shen, A. Mahmood, Y. Wang, and L. Wang, "Coordination of connected autonomous and human-operated vehicles at the intersection," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics (AIM)*, Jul. 2019, pp. 1391–1396.

[25] M. Faris, P. Falcone, and J. Sjoberg, "Optimization-based coordination of mixed traffic at unsignalized intersections based on platooning strategy," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Aachen, Germany, Jun. 2022, pp. 977–983.

[26] P. Scheffe, G. Dorndorf, and B. Alrifaee, "Increasing feasibility with dynamic priority assignment in distributed trajectory planning for road vehicles," in *Proc. IEEE 25th Int. Conf. Intell. Transp. Syst. (ITSC)*, Macau, China, Oct. 2022, pp. 3873–3879.

[27] F. Molinari, A. Katriniok, and J. Raisch, "Real-time distributed automation of road intersections," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 2606–2613, 2020.

[28] B. Chalaki and A. A. Malikopoulos, "A priority-aware replanning and resequencing framework for coordination of connected and automated vehicles," *IEEE Control Syst. Lett.*, vol. 6, pp. 1772–1777, 2022.

[29] 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)*, Denver, CO, USA, Jul. 2020, pp. 4090–4095.

[30] G. N. Bifulco, A. Coppola, A. Petrillo, and S. Santini, "Decentralized cooperative crossing at unsignalized intersections via vehicle-to-vehicle communication in mixed traffic flows," *J. Intell. Transp. Syst.*, vol. 28, no. 2, pp. 1–26, Sep. 2022.

[31] A. M. I. Mahbub and A. A. Malikopoulos, “A platoon formation framework in a mixed traffic environment,” *IEEE Control Syst. Lett.*, vol. 6, pp. 1370–1375, 2022.

[32] A. M. I. Mahbub and A. A. Malikopoulos, “Platoon formation in a mixed traffic environment: A model-agnostic optimal control approach,” in *Proc. Amer. Control Conf. (ACC)*, Atlanta, GA, USA, Jun. 2022, pp. 4746–4751.

[33] R. Hult, M. Zanon, S. Gros, H. Wymeersch, and P. Falcone, “Optimisation-based coordination of connected, automated vehicles at intersections,” *Vehicle Syst. Dyn.*, vol. 58, no. 5, pp. 726–747, May 2020.

[34] 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.

[35] 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.

[36] A. I. M. Medina, N. van de Wouw, and H. Nijmeijer, “Cooperative intersection control based on virtual platooning,” *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 6, pp. 1727–1740, Jun. 2018.

[37] M. Conforti, G. Cornuéjols, and G. Zambelli, *Integer Programming* (Graduate Texts in Mathematics), vol. 271. Cham, Switzerland: Springer, 2014.

[38] J. Andersson, J. Åkesson, and M. Diehl, “CasADi: A symbolic package for automatic differentiation and optimal control,” in *Proc. Recent Adv. Algorithmic Differentiation*, in Lecture Notes in Computational Science and Engineering, vol. 87, 2012, pp. 297–307.

[39] P. Bonami and J. Lee, “BONMIN user’s manual,” Tech. Rep., Jul. 2011.

[40] A. Wächter and L. T. Biegler, “On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming,” *Math. Program.*, vol. 106, no. 1, pp. 25–57, 2006.

Muhammad Faris received the S.T. degree (equal to B.Eng.) in electrical engineering from Universitas Gadjah Mada, Yogyakarta, Indonesia, in 2014, and the M.Sc. degree in systems and control from Delft University of Technology, Delft, The Netherlands, in 2018. He is currently pursuing the Ph.D. degree with the Chalmers University of Technology, Gothenburg, Sweden.

His current research interests include numerical methods and constrained optimal control applied to intelligent transportation systems.

Mario Zanon (Senior Member, IEEE) received the master’s degree in mechatronics from the University of Trento, Trento, Italy, in 2010, and the Diplôme d’Ingénieur degree from Ecole Centrale Paris, Paris, France, in 2010.

After research stays at the KU Leuven, University of Bayreuth, Chalmers University, and the University of Freiburg, he received the Ph.D. degree in Electrical Engineering from the KU Leuven in November 2015. He held a post-doctoral researcher position at the Chalmers University of Technology, Gothenburg, Sweden, in 2017. In 2018, he was an Assistant Professor at the IMT School for Advanced Studies Lucca, Lucca, Italy, where he became an Associate Professor in 2021. His research interests include numerical methods for optimization, economic MPC, reinforcement learning, the optimal control and estimation of nonlinear dynamic systems, in particular for aerospace, and automotive applications.

Paolo Falcone (Member, IEEE) received the M.Sc. degree (Laurea) in computer engineering from the University of Naples Federico II, Naples, Italy, in 2003, and the Ph.D. degree in information technology from the University of Sannio, Benevento, Italy, in 2007.

He is currently a Professor with the Department of Electrical Engineering, Chalmers University of Technology, Gothenburg, Sweden, and an Associate Professor with the Engineering Department, Università di Modena e Reggio Emilia, Reggio Emilia, Italy. His research interests include constrained optimal control applied to autonomous and semiautonomous mobile systems, cooperative driving and intelligent vehicles, in cooperation with Swedish automotive industry, with a focus on autonomous driving, cooperative driving, and vehicle dynamics control.

Open Access funding provided by ‘Scuola IMT Alti Studi Lucca’ within the CRUI CARE Agreement