A Sensitivity-based Heuristic for Vehicle Priority Assignment at Intersections

Citation for the original published paper (version of record):
Faris, M., Falcone, P., Zanon, M. (2023). A Sensitivity-based Heuristic for Vehicle Priority Assignment at Intersections. IFAC-PapersOnLine, 56(2): 4922-4928.
http://dx.doi.org/10.1016/j.ifacol.2023.10.1265

Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 56-2 (2023) 4922–4928

A Sensitivity-based Heuristic for Vehicle Priority Assignment at Intersections ⋆

Muhammad Faris * **Paolo Falcone** ** **Mario Zanon** ***

* *Department of Electrical Engineering, Chalmers University of Technology, Gothenburg, Sweden (e-mail: farism@chalmers.se)*
** *Department of Electrical Engineering, Chalmers University of Technology, Gothenburg, Sweden and Dipartimento di Ingegneria "Enzo Ferrari", Universita di Modena e Reggio Emilia, Italy (e-mail: paolo.falcone@chalmers.se)*
*** *IMT School for Advanced Studies Lucca, Italy (e-mail: mario.zanon@imtlucca.it)*

Abstract: This paper presents a sensitivity-based heuristic to address the dynamic priority assignment problem of connected and autonomous vehicle (CAV) and human-driven vehicle (HDV) at traffic intersections. We exploit sensitivity analysis tools to approximatively predict the CAV's performance violation as a function of the HDV states. Such predictions are then used to decide on a crossing order that preserves optimality and feasibility despite the behavior of the HDV. The proposed algorithm is compared with the baseline first-come, first-serve (FCFS) and mixed-integer nonlinear programming (MINLP) approaches. In the closed-loop simulation, we show that the heuristic is computationally much faster than MINLP and able to retain a close-to-optimal solution, which is far better than FCFS.

Keywords: Vehicle dynamic priority assignment, sensitivity analysis

1. INTRODUCTION

Coordination between connected and autonomous vehicles (CAVs) is the key to improving traffic efficiency and maintaining safety (Rios-Torres and Malikopoulos (2017)). This is especially prominent in scenarios involving challenging, dangerous areas such as unsignalized intersections.

One of the most important tasks within this context is to determine the priority assignment or crossing order, i.e., how to schedule the occupation of the intersection among the approaching vehicles. This problem, typically formulated as *mixed-integer programming* (MIP), has been proven to be NP-hard, which may prevent real-time applications computationally intractable (Chouhan and Banda (2018)). Thus, it might no longer be a viable option, even though it yields global solutions. On the other hand, a real-time approach *first-come, first-serve* (FCFS) is far from optimal in various scenarios (Meng et al. (2018)). These motivate the use of heuristic algorithms that optimize the balance between them.

There are many past works that developed heuristics to solve the crossing order problem. A model-based sequential decision-making heuristic is developed by Campos et al. (2017). Hult et al. (2019) proposed mixed-integer quadratic problem (MIQP)-based heuristic. In Mahbub et al. (2020), Chalaki and Malikopoulos (2019), the crossing order is obtained through the timeslot optimization supported by the rule-based heuristic. Xu et al. (2020) made use of a heuristic Monte Carlo search tree to iteratively retrieve the order.

The mentioned works above did not particularly consider the change of priority (order) or *reordering* issue, which can occur dynamically in mixed-traffic cases between CAVs and *human-driven vehicles* (HDVs). Due to their inability to coordinate, the uncertain behavior of the HDVs may frequently trigger reordering, e.g., when they suddenly prefer to considerably slow down (Faris et al. (2022)). As a consequence, a heuristic algorithm is required to address the problem in a computationally efficient way.

Previous research addressing the reordering problems is mostly deployed in fully autonomous traffic. Chalaki and Malikopoulos (2022) proposed a strategy based on intersection exit time minimization of the CAVs. A similar time-based sequencing notion is used by Xiao and Cassandras (2020), where it performs reordering based on the current number of considered vehicles. In Molinari et al. (2020), a negotiation-based priority approach is applied to CAVs coordination. The rules rely on the current states of vehicles but during the auction phase, the vehicles can offer their bids to negotiate their priority.

In those works, the optimality of the solution was not evaluated with respect to the global solution from, e.g., MIP formulation. Furthermore, none of them analyze approximated variation of the nominal solution to perform reordering. To that end, derivative information, i.e., sensitivity theory is required. Previous works have exploited sensitivity in the field of CAVs' coordination problem such as in Zanon et al. (2017), Hult et al. (2020), but none of

⋆ This work was supported by the Wallenberg Artificial Intelligence, Autonomous Systems, and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

10.1016/j.ifacol.2023.10.1265

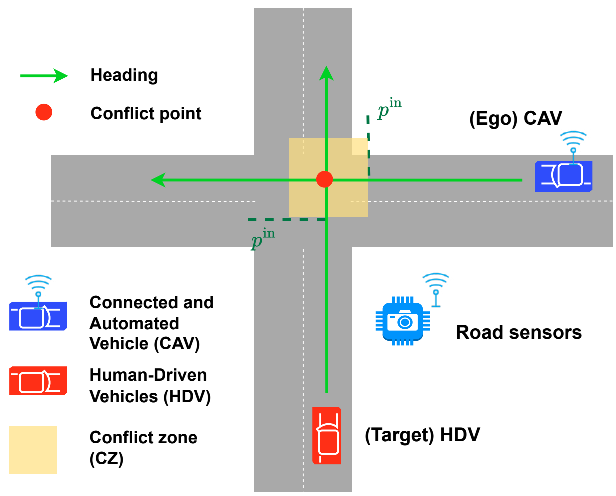

Fig. 1. An approaching HDV and CAV at unsignalized intersection

them specifically addresses the dynamic reordering problem.

In this work, we propose our contribution: A sensitivity-based heuristic approach. The sensitivity is derived from the factorization of the Karush-Kuhn-Tucker (KKT) matrix at a nominal parameter and it can be used to retrieve the approximated solution with respect to the current parameter by applying a computationally cheap first-order method. The approximated solution is then continuously monitored and investigated in the sense of its gap and velocity reference tracking as well as braking/acceleration performances to decide the crossing order dynamically.

Our ultimate goal is to eventually deploy the sensitivity-based heuristic for multi-vehicle coordination problems. Therefore, here, we start by considering a pair of ego-target vehicles' reordering problem where the number of permutations is only two. We also discuss the performance comparison of the proposed heuristic against FCFS, and MINLP in terms of solution and computational perspectives.

The paper is arranged as follows: we first introduce the intersection setting along with the vehicle model in Section 2, followed by the description of CAV optimal control problem (OCP) formulation in Section 3. In Section 4, we describe the properties of the OCP to obtain the sensitivity, used in the heuristic explained in Section 5. After that, the other methods (MINLP and FCFS) are explained in Section 6, in which we compare with the heuristic with various simulations in Section 7. Finally, the conclusion is given in Section 8.

2. MODEL DESCRIPTION

Let us consider a setting that involves a HDV (H) and a CAV (C). The vehicles travel along different directions to an *unsignalized* intersection, i.e., the conflict zone (CZ), as illustrated in Fig. 1.

Remark 1. We assume that the initial CAV's plan is yielding to the HDV, which then slows down significantly thus forcing the CAV to update its plan. The opposite scenario is not considered here for the sake of a concise presentation, yet it can be handled by the proposed approach with minor modifications. Finally, the HDV cannot communicate with other entities, yet its states and input can be measured by the CAV.

By Remark 1, a change of order (reordering) might be required at some point as keeping the current crossing order is no longer reasonable.

Definition 2. We define the two possible crossing orders for the considered setting as $\mathcal{O} = 0$ when the HDV clears the intersection before the CAV and $\mathcal{O} = 1$ when the crossing order is reversed. Thus, $\mathcal{O} = \{0, 1\}$.

2.1 Vehicle dynamics

The i-th vehicle moving along its predefined paths is described by a discrete-time, double integrator

$$x_{i,k+1} = Ax_{i,k} + Bu_{i,k},\ i \in \{\mathrm{C}, \mathrm{H}\}, \tag{1}$$

with

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

$k \in \mathbb{Z}^{+}$, where T^{p} is the prediction horizon and t^{s} is the sampling time. The state vector $x_{i,k} = [p_{i,k}\ v_{i,k}]^{\top}$ contains the longitudinal distance of the vehicle from the center of the CZ, that is, the conflict zone, and the velocity. $u_{i,k}$ is the input, i.e., acceleration/deceleration profile. Note that, other dynamic models, possibly nonlinear, can also be utilized here if needed.

Each vehicle starts from an initial condition $x_{i,0} = x_i^0,\ i \in \{\mathrm{C}, \mathrm{H}\}$. Also, their velocity and acceleration are bounded as follows.

$$u^{\min} \le u_{i,k} \le u^{\max}, \quad i \in \{\mathrm{C}, \mathrm{H}\} \tag{2a}$$
$$v^{\min} \le v_{i,k} \le v^{\max}, \quad i \in \{\mathrm{C}, \mathrm{H}\} \tag{2b}$$

with $v^{\min} > 0$, that is, vehicles do not go backward.

2.2 HDV driver model

In this work, we utilize two HDV models for prediction and simulation purposes, respectively.

Prediction The constant velocity model is employed for prediction purposes

$$u_{\mathrm{H},k} = 0, \tag{3a}$$
$$v_{\mathrm{H},k+1|t} = v_{\mathrm{H},k|t},\ \forall k \in \{0, T^{\mathrm{p}}\}, \tag{3b}$$

where $t \in \mathbb{R}^{+}$ is the current simulation time.

Simulation For simulation purposes, the reference velocity tracking model describes the HDV motion

$$u_{\mathrm{H},t+1} = k^{\mathrm{v}}(v_t^{\mathrm{ref}} - v_{\mathrm{H},t}), \tag{4}$$

where k^{v} is the gain and v_t^{ref} is the reference speed at time $t \in \mathbb{R}^{+}$.

3. PROBLEM FORMULATION

In this section, we introduce the constraint and the objective (cost) function used next to formulate the CAV motion control problem.

3.1 Side collision avoidance constraint

To prevent side collision within the CZ, the CAV's distance to the HDV is constrained as follows

$$d^{\min} \leq p_{\mathrm{H},k} - p_{\mathrm{C},k}, \tag{5}$$

where $d^{\min}$ is a minimum safety distance constant. This constraint only holds when $\mathcal{O} = 0$ and before exiting the CZ.

3.2 Objective function

An objective function is introduced for each crossing order.

Car-following objective When $\mathcal{O} = 0$, the CAV (C) follows the HDV (H) in *virtual* platooning mode (Morales Medina et al. (2018)), while minimizing the cost associated with the input over the prediction horizon T^{p}

$$J^{\mathrm{A}} = \sum_{k=0}^{T^{\mathrm{p}}-1} J_k^{\mathrm{A,p}} + J_k^{\mathrm{A,v}} + J_k^{\mathrm{A,u}}, \tag{6}$$

where

$$\begin{aligned} J_k^{\mathrm{A,p}} &= q^{\mathrm{d}}(d^{\mathrm{ref}} - (p_{\mathrm{H},k} - p_{\mathrm{C},k}))^2, \\ J_k^{\mathrm{A,v}} &= q^{\mathrm{v}}(v_{\mathrm{H},k} - v_{\mathrm{C},k})^2, \\ J_k^{\mathrm{A,u}} &= r u_{\mathrm{C},k}^2, \end{aligned}$$

and $q^{\mathrm{v}}, q^{\mathrm{d}}, r$ are the weights and d^{ref} is the reference gap. In (6), the terms $J^{\mathrm{A,x}},\ \mathrm{x} \in \{p, v\}$ penalize the deviations of the CAV's position and velocity from the safety distance to the HDV and the HDV's velocity, respectively, while the term $J^{\mathrm{A,u}}$ penalizes the CAV's control effort.

Velocity tracking objective For $\mathcal{O} = 1$, the CAV minimizes the following cost to track its reference velocity while minimizing the control effort

$$J^{\mathrm{B}} = \sum_{k=0}^{T^{\mathrm{p}}-1} q^{\mathrm{v}}(v_{\mathrm{C}}^{\mathrm{ref}} - v_{\mathrm{C},k})^2 + r u_{\mathrm{C},k}^2. \tag{7}$$

3.3 Optimal control problem (OCP)

Two optimal control problems (OCPs) are formulated for the CAV, depending on the crossing order $\mathcal{O}$.

Problem A ($\mathcal{O} = 0$) The CAV's goal is to follow the HDV, while keeping a safe distance from it, and is achieved by minimizing the cost

$$\Phi^{\mathrm{A}}(\mathbf{x}^{\mathrm{p}}, x_{\mathrm{C}}^0) = \min_{\mathbf{w}} \quad J^{\mathrm{A}}(\mathbf{w}) \tag{8a}$$

$$\text{s.t.} \quad g(\mathbf{w}) = 0,\ h(\mathbf{w}) \leq 0, \tag{8b}$$

where $\mathbf{w} = [w_1, ..., w_k, ..., w_{T^{\mathrm{p}}}]^\top \in \mathbf{R}^{n^{\mathbf{w}} \times 1}$ and $w_k = [x_{i,k}, u_{i,k}]^T$. The parameter $\mathbf{x}^{\mathrm{p}}$ contains the HDV states over the prediction horizon

$$\mathbf{x}^{\mathrm{p}} = [\mathbf{p}_{\mathrm{H},0:T^{\mathrm{p}}}]^\top,$$

while the equality and inequality constraints arising from (1), (2), (3), (5) are lumped in $g(\mathbf{w}) = 0$ and $h(\mathbf{w}) \leq 0$ as in (8b), respectively.

Problem B ($\mathcal{O} = 1$) In this case, the CAV tracks its desired speed by solving the following problem

$$\Phi^{\mathrm{B}}(x_{\mathrm{C}}^0) = \min_{\mathbf{w}} \quad J^{\mathrm{B}}(\mathbf{w}) \tag{9}$$

$$\text{s.t.} \quad \text{Eq. } (1), (2), (3),$$

where, compared to (8), the safety distance constraint (5) is dropped, as the CAV precedes the HDV. Both OCPs (8) and (9) are convex quadratic programs (QPs).

4. PROPERTIES OF THE OCP

We next recall the continuity properties of the solution of (8) with respect to the parameter $\mathbf{x}^{\mathrm{p}}$.

4.1 Karush-Kuhn-Tucker (KKT) conditions

Denote by $\mathbf{w}^*(\mathbf{x}^{\mathrm{p}}) = [w_1^*(\mathbf{x}^{\mathrm{p}}), ..., w_{T^{\mathrm{p}}}^*(\mathbf{x}^{\mathrm{p}})]^\top$ the (primal) solution of OCP (8) and by $\lambda^*(\mathbf{x}^{\mathrm{p}}),\ \mu^*(\mathbf{x}^{\mathrm{p}})$ the corresponding dual variables, with $z^* = [\mathbf{w}^*, \lambda^*, \mu^*]^\top$. The primal and dual optimal variables satisfy the following KKT conditions

$$r(z^*, \mathbf{x}^{\mathrm{p}}) = \begin{bmatrix} \nabla_w \mathcal{L}(z^*, \mathbf{x}^{\mathrm{p}}) \\ g(\mathbf{w}^*) \\ h_{\mathcal{A}}(\mathbf{w}^*, \mathbf{x}^{\mathrm{p}}) \end{bmatrix} = 0, \tag{10}$$

where

$$\mathcal{L} = J(\mathbf{w}^*, \mathbf{x}^{\mathrm{p}}) + \lambda^\top g(\mathbf{w}^*) + \mu_{\mathcal{A}}^\top h_{\mathcal{A}}(\mathbf{w}^*, \mathbf{x}^{\mathrm{p}}).$$

4.2 Continuity and differentiability properties

The following assumptions are used next to provide a continuity result (Still (2018)) for the parametric solution of (8)

Assumption 1. $\mathbf{w}^*$ is a regular point. Thus, *linear independence constraint qualification* (LICQ) (Still (2018)) holds within the neighborhood of $\mathbf{x}^{\mathrm{p}}$.

Assumption 2. Strict *second order sufficient condition* (SOSC) (Still (2018)) holds at the solution.

Proposition 3. If Assumptions 1 and 2 hold at $\mathbf{x}^{\mathrm{p}}$, then the parametric solution $\mathbf{w}^*(\mathbf{x}^{\mathrm{p}})$ is continuous in a neighborhood of $\mathbf{x}^{\mathrm{p}}$. Furthermore, if there are no *weakly active* inequality constraints, then $\nabla_{\mathbf{x}^{\mathrm{p}}} \mathbf{w}^*(\mathbf{x}^{\mathrm{p}})$ exists.

By leveraging Proposition 3, the *approximated* parametric solution of (8), $\hat{\mathbf{w}}^*(\mathbf{x}^{\mathrm{p}})$, can be calculated based on the sensitivities as shown next. In fact, if the continuity and differentiability conditions hold for the solution of (8), $\hat{\mathbf{w}}^*(\mathbf{x}^{\mathrm{p}})$ can be obtained using a linear, first-order Taylor approximation (Still (2018))

$$\hat{\mathbf{w}}^*(\mathbf{x}^{\mathrm{p}}) \approx \mathbf{w}^*(\tilde{\mathbf{x}}^{\mathrm{p}}) + \frac{\partial \mathbf{w}^*(\mathbf{x}^{\mathrm{p}})}{\partial \mathbf{x}^{\mathrm{p}}}(\mathbf{x}^{\mathrm{p}} - \tilde{\mathbf{x}}^{\mathrm{p}}). \tag{11}$$

The approximated states $\hat{p}_{i,k}(\mathbf{x}^{\mathrm{p}}), \hat{v}_{i,k}(\mathbf{x}^{\mathrm{p}})$, and input $\hat{u}_{i,k}(\mathbf{x}^{\mathrm{p}})$ can thus be extracted from $\hat{w}_k^*(\mathbf{x}^{\mathrm{p}})$.

The sensitivities in (11) can be retrieved from the derivatives of KKT conditions (10)

$$\frac{\partial z}{\partial \mathbf{x}^{\mathrm{p}}} = -\left(\frac{\partial r(z, \mathbf{x}^{\mathrm{p}})}{\partial z}\right)^{-1} \frac{\partial r(z, \mathbf{x}^{\mathrm{p}})}{\partial \mathbf{x}^{\mathrm{p}}}, \tag{12}$$

where $\frac{\partial \mathbf{w}}{\partial \mathbf{x}^{\mathrm{p}}}$ are in the first $n^{\mathbf{w}}$ rows of $\frac{\partial z}{\partial \mathbf{x}^{\mathrm{p}}}$.

5. HEURISTIC ALGORITHM

In the considered setting, the problem of optimal crossing order assignment is typically formulated as a mixed-integer nonlinear programming (MINLP), as explained later in Section 6.2. In the literature above, it has been mentioned that solving a MINLP is computationally expensive. As an alternative, we propose a heuristic method.

The heuristic's task is to decide the crossing order $\mathcal{O}$ at time $t \in \mathbb{R}^+$, depending on the vehicles' states and parameter $\mathbf{x}^{\mathrm{p}}$. Once decided, the CAV solves either problem (8) or (9). The idea underlying the heuristic is to switch crossing order in the case where, for the current order, the HDV's behavior is leading to an increase of the cost (performance) function beyond a prescribed bound. This notion will be explained in detail next.

Assume that the current crossing order is $\mathcal{O} = 0$ and Problem A has been solved at $t = 0$, yielding solution $\mathbf{w}^*(\tilde{\mathbf{x}}^{\mathrm{p}})$. For the next timesteps $t > 0$, the cost J^{A} can be approximated by evaluating (6) at $\hat{\mathbf{w}}^*(\mathbf{x}^{\mathrm{p}})$, cheaply calculated as in (11) for the current $\mathbf{x}^{\mathrm{p}}$. In this way, the cost can be monitored in order to detect its growth beyond the following bounds

$$J_k^{\mathrm{A,p}}(\hat{p}_{i,k}(\mathbf{x}^{\mathrm{p}})) \leq q^{\mathrm{d}}||d^{\mathrm{ub}}||^2, \ \forall k \in \{0, T^{\mathrm{p}}\}, \tag{13a}$$

$$J_k^{\mathrm{A,v}}(\hat{v}_{i,k}(\mathbf{x}^{\mathrm{p}})) \leq q^{\mathrm{v}}||v^{\mathrm{ub}}||^2, \ \forall k \in \{0, T^{\mathrm{p}}\}, \tag{13b}$$

$$J_k^{\mathrm{A,u}}(\hat{u}_{i,k}(\mathbf{x}^{\mathrm{p}})) \leq q^{\mathrm{d}}||u^{\mathrm{lb}}||^2, \ \forall k \in \{0, T^{\mathrm{p}}\}, \tag{13c}$$

where the bounds d^{ub}, v^{ub}, u^{lb} are chosen to reflect the maximum allowed gap, velocity tracking error from the *target* vehicle, and CAV's breaking/acceleration efforts. These performance inequalities, over T^{p}, can be compactly written as $J^{\mathrm{A}} \leq J^{\mathrm{ub}}$.

In Algorithm 1, the growth of the cost beyond each of the bounds in (13) is continuously monitored and counted by $s \in \mathbf{R}^{3\times 1}$ (line 9 of the algorithm). s collects the number of occasions every time a violation $J^{\mathrm{A}} > J^{\mathrm{ub}}$ (line 8) is detected. A *consistency* check is further implemented to avoid *false positive* situations, due to, e.g., noise and prevent an early change of order. This notion can be seen in line 14, where if there is no violation detected in the current simulation time $t \in \mathbb{R}^+$, but the counter $s = 1$ due to a violation in $t-1$, we reset it back to zero (line 15). Reordering occurs if any of the counters shows n^{max} consecutive violations (lines 12-14). The order is frozen once the HDV has been close to CZ (line 6).

Algorithm 1 Sensitivity-based Heuristic

Input: $x_{\mathrm{C},k}, \ \mathbf{x}^{\mathrm{p}}, \ \mathcal{O} = 0$
Output: $\mathcal{O}$
1: Select $\mathcal{O} = 0, \ \Phi = \Phi^{\mathrm{A}}$, set $s = 0$ ▷ Initialization
2: Solve QP $\Phi^{\mathrm{A}}(\tilde{\mathbf{x}}^{\mathrm{p}})$
3: Compute sensitivities $\frac{\partial \mathbf{w}}{\partial \mathbf{x}^{\mathrm{p}}}$
4: **for** $t \in \mathbb{R}^+$ **do**
5: Obtain $x_{\mathrm{C},t}$ and new $\mathbf{x}^{\mathrm{p}}$
6: **if** $p_{\mathrm{H},t} \leq p^{\mathrm{in}} - \delta$ **then**
7: Compute $J^{\mathrm{A}}(\hat{\mathbf{w}}(\mathbf{x}^{\mathrm{p}}))$
8: **if** $J^{\mathrm{A}} > J^{\mathrm{ub}}$ **then** ▷ Performance bounds
9: $s = s + 1$ ▷ Counting the violation
10: **end if**
11: **end if**
12: **if** $s == n^{\mathrm{max}}$ **then** ▷ Consistency check
13: Set $\mathcal{O} = 1, \ \Phi = \Phi^{\mathrm{B}}$,
14: **else if** $s == 1$ && $\forall k: \ J^{\mathrm{A}} \leq J^{\mathrm{ub}}$ **then**
15: $s = 0$ ▷ Reset if it was a false sign
16: **end if**
17: **end for**

6. OTHER METHODS

The proposed heuristic is compared against other methods below.

6.1 First-Come-First-Served (FCFS)

FCFS sorts the crossing order based on the vehicle position when entering the coordination area with respect to the center of the intersection or the time they arrive at the intersection. Due to Remark 1, FCFS starts and fixes its order at $\mathcal{O} = 1$. This method serves as a baseline for the heuristic and MINLP.

6.2 Mixed-Integer nonlinear Programming (MINLP)

MINLP presents the original optimal solution where the crossing order is obtained through generic solvers. It is essentially a combination of OCP (8) and (9), that implements the switching safe distance constraint

$$\begin{cases} p_{\mathrm{H},k} - p_{\mathrm{C},k} - d^{\mathrm{min}} \geq 0 & \text{if} \quad \mathcal{O} == 0 \\ \emptyset & \text{else if } \mathcal{O} == 1. \end{cases} \tag{14}$$

and the switching objective function

$$J^{\mathrm{U}} = \begin{cases} J^{\mathrm{A}} & \text{if} \quad \mathcal{O} == 0 \\ J^{\mathrm{B}} & \text{else if } \mathcal{O} == 1. \end{cases} \tag{15}$$

Both imply MIP formulation, which is non-convex. Based on Remark 1, the safe distance constraint only applies when $\mathcal{O} = 0$.

Remark 4. Constraint (14) cannot be directly implemented, as the solver naturally chooses the unconstrained case. Therefore, an *auxiliary* constraint is added

$$\begin{cases} p_{\mathrm{H},k} - p_{\mathrm{C},k} - d^{\mathrm{min}} + \eta_k \geq 0, & \text{if} \quad \mathcal{O} == 0 \\ p_{\mathrm{C},k} - p_{\mathrm{H},k} + d^{\mathrm{com}} + \eta_k \geq 0, & \text{else if} \quad \mathcal{O} == 1, \end{cases} \tag{16}$$

where the second constraint represents situations where the CAV overtakes the HDV. $0 < d^{\mathrm{com}} < d^{\mathrm{min}}$ is a constant to compensate for the safety gap and η_k is a slack variable to avoid infeasibility during switching.

The overall MINLP problem is

$$\begin{aligned} \Phi^{\mathrm{U}}(x_{\mathrm{C}}^0, \mathbf{x}^{\mathrm{p}}) = \ & \min_{\mathbf{w}} \quad J^{\mathrm{U}}(\mathbf{w}) + q^{\mathrm{s}}||\eta_{0:T^{\mathrm{p}}}||_2^2 \\ & \text{s.t.} \quad \text{Eq. } (1), (2), (3), (16), \\ & \qquad\quad \mathcal{O} \in \{0, 1\}, \end{aligned} \tag{17}$$

where q^{s} is the penalty weight. The corresponding algorithm is written in Algorithm 2. Note that we also apply an additional check (line 3 in the algorithm) to make sure that the MINLP does not change the order when it is too close to the CZ (line 7).

7. NUMERICAL SIMULATIONS

In this section, the proposed heuristic is compared against the FCFS and MINLP methods in closed-loop simulation with different experiments. Evaluation includes optimality and computational aspects. Furthermore, we study the susceptibility of the heuristic with different objective bounds. In the simulations, we set $d^{\mathrm{min}} = 4$ m and $d^{\mathrm{ref}} = 6$ m. The input is constrained between ± 2 m/s^2. The center of CZ is the zero position, with $p^{\mathrm{in}} = -4$. In line with Remark 1, the HDV is placed *virtually* in front of the CAV

Algorithm 2 Mixed-Integer nonlinear Programming (MINLP)

Input: $x_{\mathrm{C},k},\ \mathbf{x}^{\mathrm{p}}$
Output: $\mathcal{O}$
for $t \in \mathbb{R}^{+}$ **do**
  Obtain $x_{\mathrm{C},t}$ and new $\mathbf{x}^{\mathrm{p}}$
  if $p_{\mathrm{H},t} \leq p^{\mathrm{in}} - \delta$ **then**
    Solve MINLP problem Φ^{U}
    Obtain $\mathcal{O} = \arg\min \Phi^{\mathrm{U}}$
  else
    Keep the same $\mathcal{O}$
  end if
end for

and both have similar initial speeds. Accordingly, we set $\mathcal{O} = 0$ for FCFS and heuristic at the beginning. Also, MINLP and heuristic no longer consider reordering when the order has switched to $\mathcal{O} = 1$. The controller weights are all set to unity. All simulations are carried out using MATLAB with Casadi framework, on a laptop with Intel Core-i5 processor and 16 GB of RAM. The QP problems are solved with Ipopt, while Bonmin is used for solving the MINLP.

7.1 Performance evaluation

The results presented next are obtained by simulating the HDV with the model (4), while model (3) is utilized by the three approaches. The HDV's speed profile resembles a braking (deceleration) maneuver. This is achieved by choosing the following two reference speed profiles, $v_t^{\mathrm{ref}} = 36$ km/h, $t = [0, 3.3]$ seconds (s) and $v_t^{\mathrm{ref}} = 22$ km/h, $t = [3.4, 10]$ s. Normally distributed noise Δu_k with zero mean and $\sigma = 0.1$, is added to the input of the HDV's simulation model to mimic the randomness introduced by a human driver. The heuristics bounds $d^{\mathrm{ub}}, v^{\mathrm{ub}}, l^{\mathrm{lb}}$ are set to 0.5 m, 2.0 km/h, -1.2 m/s^2, respectively, and $n^{\max} = 3$. We simulate 50 experiments, each with random bounds on the HDV input, uniformly distributed within 10% of the nominal bounds. The rest of the constants remain unchanged. For evaluation, we consider τ, that is, the time (in seconds) when the change of order occurs. Moreover, the total objective value is the sum of the values of the entire simulation time, while the computational performance is measured from the average time required to decide the order in each time t. The mean, normalized mean (from the MINLP as the global optimal solution), and standard deviation (std.) are then calculated from the results of all scenarios.

Fig. 2 shows an example from the first experiment. On the left, we see that both heuristic (Heur.) and MINLP are able to tell the CAV to takeover the HDV before they are reaching the CZ. Both have almost the same trajectories, where the MINLP is slightly ahead. This is due to 0.1 second difference in reordering of the heuristic, as indicated by Fig. 3. The difference might be caused by the choice of the performance bounds or $n^{\max}$, as explained further in the next subsection. In Table 1, we can see that the difference in the mean reordering time τ between the MINLP and the heuristic is very small, implying that the heuristic solution is close to MINLP. A similar conclusion can be drawn by comparing the mean and normalized mean objectives J^{A} of the heuristic and the MINLP, respectively. From the table, we see that the heuristic standard deviations of τ and J^{A} are slightly higher than those of MINLP, yet much smaller than those of FCFS. By comparing the costs of the heuristic and the MINLP against the FCFS, it is clear that keeping the CAV behind the HDV is clearly disadvantageous.

The advantage of the heuristic is shown in the average computation time in Table 1, where it demonstrates faster processing time than MINLP. This is expected from utilizing a simple yet efficient first-order method compared to solving a MINLP problem. FCFS is clearly the fastest, as it does not change the order. In terms of the standard deviation of the computational performance, the heuristic shows better consistency than MINLP.

7.2 Susceptibility of the objective bounds

One of the drawbacks of the heuristic is the bounds in (13) which requires a tuning process. In this section, we shed some light on how these tuning parameters affect the performance of the heuristic.

In the previous subsection, the heuristic manages to attain close-to-optimal solutions thanks to the carefully selected values of the objective bounds in (13). We next study the impact of the variations of these bounds when their values are slightly varied, as shown in Table 2. For each case, we simulate the behavior of the heuristics in the same 50 experiments as before.

When the position bound d^{ub} is set to 0.4, the reordering averagely occurs much earlier implying a lower cost value than the previous tuning and also the MINLP. This also happens with the other two bounds v^{ub} and u^{lb} when they are tightened. In real scenarios, early reordering might not be preferable as it could be unnecessarily induced by the HDV. However, we observe in the normalized mean of the objective values (with respect to the MINLP results) that, when they are tightened, their solutions are not far from the MINLP and the difference between them can be considerably small. Their standard deviations (std.) are very small as well.

Next, when d^{ub} is decreased to 0.6, the reordering occurs much later, followed by a higher objective value. A similar pattern is also seen with the other two bounds when they are loosened. A striking result is indicated by the velocity bound ($v^{\mathrm{ub}} = 2.5$ km/h) that the reordering difference is much larger than the others, and it is also the case for the standard deviation. In the normalized objective value, the loosen velocity bound also shows the largest deviation, far higher than the other bounds. Hence, using velocity bound as a performance indicator with sensitivity is the least preferable compared to the others, as it is the most sensitive to change, even though its standard deviation is the lowest when it is tight. In contrast, u^{lb} proves to be the most reliable term by having the lowest normalized value both when it is tightened and loosened.

Overall, we observe that to retrieve results similar to the MINLP, the bounds on the states tracking deviation have to be relatively close to the references, while the input bound can be closer to the constraint.

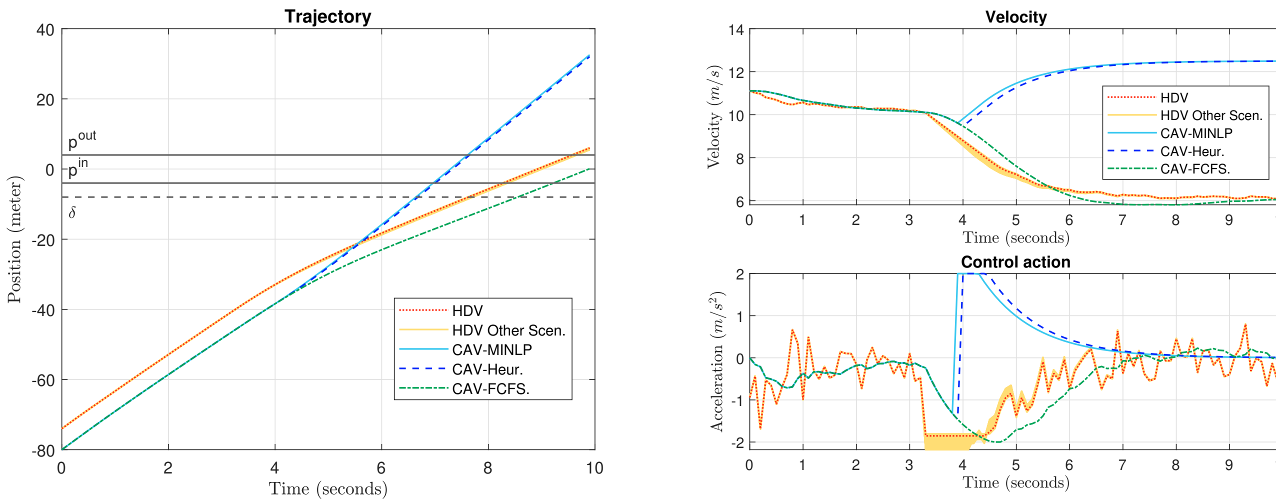

Fig. 2. Position (left), velocity, and acceleration profile (right) of the HDV and the CAV from the first experiment, except the yellow lines which are the HDV positions, velocities, and inputs from the rest of the experiments. The yellow lines show variations in the HDV behavior, particularly noticeable during deceleration at $t = 3.4$ s.

Table 1. Performance evaluation result.

Methods	τ (sec.)		Total objective value J^A			Avg. comp. time (sec.)	
	Mean	Std.	Mean	Normalized mean (%)	Std.	Mean	Std.
MINLP	3.87	0.045	622.76	N/A	27.91	0.1503	0.01400
Heuristics	3.94	0.049	682.07	8.84	33.20	0.0593	0.00300
FCFS	N/A	N/A	1056.50	68.45	41.23	0.0003	0.00001

Table 2. Heuristic bounds susceptibility results.

Bounds		d^{ub}		v^{ub}		u^{lb}	
		0.4	0.6	1.5	2.5	-1.0	-1.4
τ (sec.)	Mean	3.80	4.08	3.81	4.20	3.84	4.05
	Std.	0.00	0.04	0.02	0.11	0.04	0.05
Total objective value	Mean	567.46	834.20	571.76	1012.40	595.48	806.92
	Normalized mean (%)	8.65	34.01	7.96	62.5	4.21	29.74
	Std.	10.05	36.47	16.51	128.63	28.72	38.41

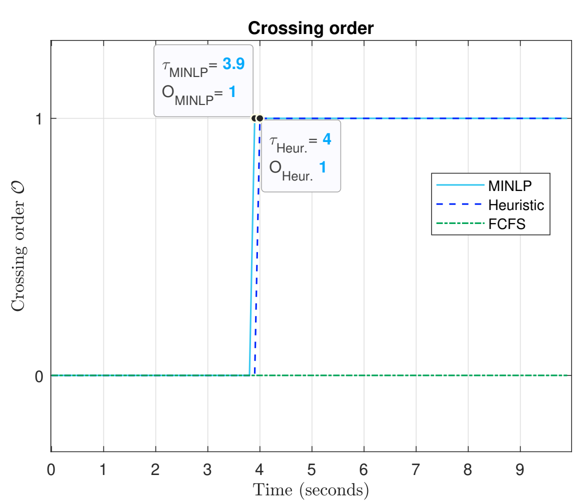

Fig. 3. Crossing order (priority) of the first experiment.

8. CONCLUSION

This work addresses the development of a computationally efficient heuristic algorithm for the dynamic priority assignment problem of an ego CAV against a target HDV and in an unsignalized intersection based on the sensitivity information.

The sensitivity utilized by the first-order Taylor method enables the approximation of solution with respect to the current parameter, that is, the HDV states in a computationally inexpensive way. The estimated solution is then investigated through cost performance evaluation to dynamically decide the crossing order. The proposed heuristic is compared with FCFS and MINLP in closed-loop experiments with different HDV input bounds. The heuristic manages to achieve close-to-optimal solutions, similar to that of MINLP but with a cheaper computation effort. Furthermore, the susceptibility study highlights the performance of each performance bound, where the input bound outperforms the others in terms of reliability.

A natural extension of this work is to address dynamic priority assignment in mixed-traffic, multi-vehicle coordination problems, where the number of permutations is much higher. To ensure sensible performance, the current heuristic needs to be further developed.

REFERENCES

Campos, G.R., Falcone, P., Hult, R., Wymeersch, H., and Sjöberg, J. (2017). Traffic Coordination at Road

Intersections: Autonomous Decision-Making Algorithms Using Model-Based Heuristics. *IEEE Intelligent Transportation Systems Magazine*, 9(1), 8–21.

Chalaki, B. and Malikopoulos, A.A. (2019). An Optimal Coordination Framework for Connected and Automated Vehicles in two Interconnected Intersections. In *IEEE Conference on Control Technology and Applications (CCTA)*, 6. IEEE, Hong Kong.

Chalaki, B. and Malikopoulos, A.A. (2022). A Priority-Aware Replanning and Resequencing Framework for Coordination of Connected and Automated Vehicles. *IEEE Control Systems Letters*, 6, 1772–1777.

Chouhan, A.P. and Banda, G. (2018). Autonomous Intersection Management: A Heuristic Approach. *IEEE Access*, 6, 53287–53295.

Faris, M., Falcone, P., and Sjoberg, J. (2022). Optimization-Based Coordination of Mixed Traffic at Unsignalized Intersections Based on Platooning Strategy. In *2022 IEEE Intelligent Vehicles Symposium (IV)*, 977–983. IEEE, Aachen, Germany.

Hult, R., Zanon, M., Gras, S., and Falcone, P. (2019). An MIQP-based heuristic for Optimal Coordination of Vehicles at Intersections. In *Proceedings of the IEEE Conference on Decision and Control*, volume 2018-December, 2783–2790. ISBN: 9781538613955.

Hult, R., Zanon, M., Gros, S., Wymeersch, H., and Falcone, P. (2020). Optimisation-based coordination of connected, automated vehicles at intersections. *Vehicle System Dynamics*, 58(5), 726–747, publisher: Taylor & Francis.

Mahbub, A.M., Malikopoulos, A.A., and Zhao, L. (2020). Decentralized optimal coordination of connected and automated vehicles for multiple traffic scenarios. *Automatica*, 117, 108958.

Meng, Y., Li, L., Wang, F.Y., Li, K., and Li, Z. (2018). Analysis of Cooperative Driving Strategies for Nonsignalized Intersections. *IEEE Transactions on Vehicular Technology*, 67(4), 2900–2911.

Molinari, F., Katriniok, A., and Raisch, J. (2020). Real-Time Distributed Automation Of Road Intersections. *IFAC-PapersOnLine*, 53(2), 2606–2613.

Morales Medina, A.I., Wouw, N.V.D., and Nijmeijer, H. (2018). Cooperative Intersection Control Based on Virtual Platooning. *IEEE Transactions on Intelligent Transportation Systems*, 19(6), 1–14.

Rios-Torres, J. and Malikopoulos, A.A. (2017). A Survey on the Coordination of Connected and Automated Vehicles at Intersections and Merging at Highway On-Ramps. *IEEE Transactions on Intelligent Transportation Systems*, 18(5), 1066–1077, publisher: IEEE.

Still, G. (2018). Lectures on Parametric Optimization: An Introduction. *Optimization Online*.

Xiao, W. and Cassandras, C.G. (2020). Decentralized Optimal Merging Control for Connected and Automated Vehicles with Optimal Dynamic Resequencing. *Proceedings of the American Control Conference*, 2020-July, 4090–4095, iSBN: 9781538682661.

Xu, H., Zhang, Y., Li, L., and Li, W. (2020). Cooperative Driving at Unsignalized Intersections Using Tree Search. *IEEE Transactions on Intelligent Transportation Systems*, 21(11), 4563–4571.

Zanon, M., Gros, S., Wymeersch, H., and Falcone, P. (2017). An Asynchronous Algorithm for Optimal Vehicle Coordination at Traffic Intersections. *IFAC-PapersOnLine*, 50(1), 12008–12014.