Optimal Control of Automated Vehicles Crossing a Lane-free Signal-free Intersection*

Mehdi Naderi, Panagiotis Typaldos, Markos Papageorgiou, *Life Fellow, IEEE*

***Abstract*— Developing signal-free intersections, where connected automated vehicles (CAVs) for all OD (origin-destination) movements are appropriately guided to cross simultaneously, may significantly improve throughput and reduce fuel consumption. Naturally, vehicles in the intersection area are not bound to lanes; therefore, it is reasonable to consider the crossing area as a lane-free infrastructure for further improved exploitation. This paper proposes a joint optimal control approach for CAVs crossing signal-free and lane-free intersections. Specifically, the control inputs of all vehicles, comprising acceleration and steering angle, are optimized over a time-horizon by solving a single optimal control problem (OCP) based on the bicycle model of vehicle dynamics. The cost function includes proper terms to ensure smooth and collision-free motion, while also considering fuel consumption and desired-speed tracking, when possible. Appropriate constraints are designed to respect the intersection boundaries and ensure smooth vehicle movements towards their respective destinations. The defined OCP is solved numerically via an efficient Feasible Direction Algorithm (FDA), which is acceptably fast. A challenging demonstration example confirms the effectiveness of the suggested method.**

I. Introduction

Over the last few decades, many companies and research institutes around the world have been working on a variety of vehicle automation and communication systems, which may be leveraged to tackle problems caused by traffic congestion like delays, air pollution, and traffic safety degradation [1]-[2]. In particular, employing vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications as well as various sensors, high-automation or virtually driverless connected automated vehicles (CAVs), that assess their surrounding area and make pertinent driving decisions according to well-designed control strategies, have been tested in real traffic conditions [3].

Recently, the TrafficFluid concept, a novel paradigm for vehicular traffic, which applies at high levels of vehicle automation and communication was proposed in [4], relying on two combined principles: (a) Lane-free traffic, whereby vehicles are not bound to fixed traffic lanes, as in conventional traffic, but may drive anywhere on the 2-D surface of the road; and (b) Vehicle nudging, whereby vehicles may influence the movement of other vehicles on the side or in front of them. Over the last couple of years, several movement strategies were proposed for CAVs on diverse lane-free infrastructures, under the TrafficFluid paradigm, using different methodologies including: ad-hoc strategies [4],[5], optimal control [6], [7], reinforcement learning [8], nonlinear feedback control [9]; see [10] for a brief review. In particular, lane-free driving on large-scale urban roundabouts was considered [11]-[13]among other road infrastructure types.

Intersections are latent bottlenecks for traffic flow in urban road networks and account for the vast majority of vehicle stops, which incur additional fuel consumption and emissions. The primary problem is due to conflicts among vehicles with antagonistic movements, that is conventionally addressed via traffic signals, whose proper management has been researched for decades [1], [14]. In view of the emerging CAV technology, there has been increasingly interest in signal-free intersection operation to increase throughput and reduce vehicle stops; and many control approaches for safe CAV crossing through signal-free intersections were proposed, see some review papers [15]-[17]. Those works consider lane-based traffic, even within the intersection area, assuming pre-specified crossing paths for vehicles, with a limited number of conflict points among paths. Then they specify a conflict-free crossing solution for all vehicles based on different strategies, like FIFO (first-in-first-out) or optimization-based crossing sequence, guiding vehicles such that they never pass a common conflict point simultaneously. It is interesting to note that optimization-based approaches involve discrete variables to handle conflict avoidance, which may lead to scaling problems if many vehicles need to be considered simultaneously.

More recently, some lane-free or path-free approaches were proposed, e.g. in [18]-[25], in which vehicles may drive on any position of the intersection's 2-D surface, instead of moving along pre-determined paths. This may further improve the intersection throughput because of more flexible exploitation of the available space. In a sequence of developments, see e.g. [18]-[21], Li, et al. employ the bicycle kinematic model for the vehicle dynamics and formulate an optimal control problem (OCP) with specific initial and final conditions and free or fixed final time, aiming to maximize vehicle advancement and hence intersection throughput. Constraints are defined to ensure the physical limits for speed, acceleration, steering angle, and its change rate; as well as collision avoidance, the latter by approximating vehicle shapes and road boundaries with circles. Thus, the OCP solution provides conflict-free vehicle trajectories, which are flexibly distributed in the whole intersection area. However, despite involving only real-valued

* The research leading to these results has received funding from the European Research Council under the European Union's Horizon 2020 Research and Innovation Programme / ERC Grant Agreement no. 833915, project TrafficFluid, see: https://www.trafficfluid.tuc.gr

All authors are with Dynamic Systems and Simulation Laboratory (DSSL), Technical University of Crete, Chania, Chania, Greece. (e-mail: {mnaderi, ptypaldos, mpapageorgiou}@tuc.gr); M. Papageorgiou is also with the Faculty of Maritime and Transportation, Ningbo University, Ningbo, China.

variables, the OCP numerical solution is not real-time feasible, hence several simplifications are introduced to reduce the computational burden and facilitate real-time implementation. In [21], small disjoint vehicle 'batches' are introduced that cross the intersection one at a time, and a small-scale OCP is considered for each batch while crossing; the batch crossing order being managed via a FIFO policy.

Amouzadi et al. [22],[23] suggested another optimal control approach for signal-free and lane-free intersection crossing, based also on the bicycle model of vehicle dynamics. The method minimizes the final time, at which all handled vehicles reach their pre-selected final positions in the respective exit branches; while imposing physical limits as constraints. Also, using polytopes to represent vehicles and road boundaries, non-convex constraints are introduced to avoid collisions and boundary violation, which are then converted to convex ones. The defined OCP is solved in open-loop mode, while online application is differed to future work. The reported required computation time seems relatively high when many vehicles are handled. Also, considering the same exit time for all vehicles may lead to unnecessary deceleration for some vehicles.

Finally, Ahmadi and Carlson [24] utilize an optimal control approach, which had been presented in [25], to develop a Model Predictive Control (MPC) scheme, to navigate vehicles crossing a signal-free and lane-free intersection. The method aims to maximize vehicle progress while also accounting for the control effort. Problem constraints reflect physical limits. The considered discrete-time version of the bicycle model is not exact but derived from Euler discretization, which may call for a sufficiently small time-step to mitigate numerical errors. Also, using a circle (rather than ellipse) in collision avoidance constraints may waste lateral space. Finally, the used time-horizon of 0.75 s seems short.

In this work, we propose a new OCP formulation for vehicles crossing a signal-free and lane-free intersection without consideration of limiting assumptions, as specific initial and final conditions that may reduce solution flexibility. Appropriate terms in the cost function encourage vehicles to safely, smoothly, and efficiently cross the intersection. Wide Origin-Destination (OD) corridors are defined, and appropriate constraints guarantee that vehicles remain therein. Furthermore, physical limits are considered, in particular also for the vehicles' centrifugal accelerations to ensure passenger convenience. The OCP is solved by an efficient Feasible Direction Algorithm (FDA) [26].

The rest of the paper is organized as follows. Vehicle dynamics are presented in Section II. OD corridors, boundary controllers and desired orientations are explained in Section III. Section IV describes the OCP and its numerical solution. Demonstration results are presented in Section V (along with a video), and concluding remarks are given in Section VI.

II. Vehicle Modeling

A. Vehicle Dynamics

In view of frequent and strong turnings in the intersection, it is necessary to employ a sufficiently accurate model to represent vehicle dynamics, such as the kinematic bicycle model (see Fig. 1), which is described by the state equations [9]

$$\begin{aligned} \dot{x} &= v\cos(\theta) \\ \dot{y} &= v\sin(\theta) \\ \dot{\theta} &= \sigma^{-1} v\tan(\delta) = \mu \\ \dot{v} &= F \end{aligned} \tag{1}$$

where σ is the vehicle length; x and y are the longitudinal and lateral position coordinates of the vehicle's rear axle midpoint; v is the vehicle speed; and θ is its orientation. The origin of coordinates is the center of the intersection. The model has two control inputs: acceleration F and steering angle δ. For the sake of simplicity, we define an intermediary control variable $\mu = \sigma^{-1} v\tan(\delta)$, which replaces the steering angle δ.

B. Transformation for Rotated (Skewed) Coordinates

Since each intersection has several arms with different orientations, to have a uniform methodology for all of them, we may rotate the coordinates by an angle $\theta' \in [0, 2\pi)$, which may be the orientation of any intersection branch. The transformation is illustrated in Fig. 2, where the rotated (skewed) coordinates are (x', y') and can be derived as [13]:

$$\begin{aligned} x' &= x\cos\theta' + y\sin\theta' \\ y' &= y\cos\theta' - x\sin\theta' \end{aligned} \tag{2}$$

The difference between the vehicle orientation and the skewed angle θ' is considered as a new state variable $\xi = \theta - \theta'$ (skewed orientation). Calculating the time-derivatives of the new state variables yields the state equations

$$\begin{aligned} \dot{x}' &= v\cos\xi \\ \dot{y}' &= v\sin\xi \\ \dot{\xi} &= \mu \\ \dot{v} &= F \end{aligned} \tag{3}$$

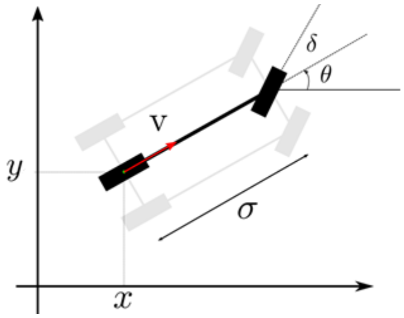

Figure 1. The bicycle model details [9]

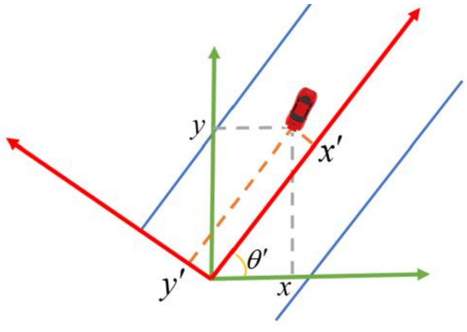

Figure 2. Transformation for rotated coordinates [13]

C. *Sampled-Data Bicycle Model*

The bicycle model needs to be discretized for its implementation in practical frameworks, like simulators or discrete-time OCPs. The exact sampled-data model is obtained through analytical integration of (1), while considering constant values for the control inputs during a sampling period [27], which yields

$$x(k+1) = x(k) + \frac{\sigma}{\tan(\delta(k))} *$$

$$\left(\sin\left(\theta(k) + v(k)\frac{\tan(\delta(k))}{\sigma}T + F(k)\frac{\tan(\delta(k))}{2\sigma}T^2 \right) - \sin\left(\theta(k)\right) \right)$$

$$y(k+1) = y(k) + \frac{\sigma}{\tan(\delta(k))} *$$

$$\left(\cos\left(\theta(k)\right) - \cos\left(\theta(k) + v(k)\frac{\tan(\delta(k))}{\sigma}T + F(k)\frac{\tan(\delta(k))}{2\sigma}T^2 \right) \right)$$

$$\theta(k+1) = \theta(k) + v(k)\frac{\tan(\delta(k))}{\sigma}T + F(k)\frac{\tan(\delta(k))}{2\sigma}T^2$$

$$v(k+1) = v(k) + F(k)T \qquad (4)$$

where T is the sampling period. Note that, strictly speaking, model (4) is valid if $\delta \neq 0$, which can be assumed permanently valid if we replace zero with an arbitrarily small positive value without any noticeable consequences for the modelling accuracy.

III. OD Corridors and Desired Orientation

In this section, the basic frame for vehicles crossing an intersection is described. First, overlapping OD corridors are designed, each corridor determining the allowable intersection area for vehicles with the corresponding OD. Second, boundary controllers are developed to guarantee that vehicles do not exceed their associated OD corridor. Finally, position-dependent desired orientations are determined for each OD corridor. The exposition considers a perpendicular four-arm intersection, as usual in signal-free intersection works, involving all (twelve) OD-movements of vehicles. However, the developed approach is general enough to apply to any intersection structure.

A. *OD Corridors*

To keep the vehicles within the intersection boundaries, avoid sudden changes in their movements, and guarantee their arrival at their respective destinations, we define suitable overlapping OD corridors in which vehicles with a specific intention (straight, left or right) are allowed to move. The corridors should be wide, to allow for better exploitation of the available space in the lane-free context. There are many possibilities for corridor determination, but, for simple and systematic design, we employ a combination of straight and circular corridor boundaries that also ensure smooth motion even if a vehicle moves on the boundary. Note that vehicles are allowed to drive on the corridor boundaries, something that may be beneficial, since it may leave more space for other crossing vehicles and improve the exploitation level of infrastructure. We present the corridor design according to the vehicle intention (straight, left or right) at some entering branch, and the method can be applied to all four intersection branches to obtain a total of twelve overlapping OD corridors.

TABLE I. OD CORRIDOR PARAMETERS

Parameter	Description	Range or Value
L_x	Length of crossing area	
L_y	Width of crossing area	
θ_{en}	Entrance branch orientation	$\{\pi, 3\pi/2, 0, \pi/2\}$
θ_{ex}	Exit branch direction	$\{0, \pi/2, \pi, 3\pi/2\}$
W_{en}	Entrance branch width	$0.5\left\lvert L_x \sin\theta_{\text{en}} + L_y \cos\theta_{\text{en}} \right\rvert$
W_{ex}	Exit branch width	$0.5\left\lvert L_x \sin\theta_{\text{ex}} + L_y \cos\theta_{\text{ex}} \right\rvert$
$r_{\text{L,u}}$	Radius of circular boundary for the upper limit while turning left	$[0, L_y/2 + L_x]$
$\mathbf{c}_{\text{L,u}}$	Center of circular boundary for the upper limit while turning left	$\begin{pmatrix} r_{\text{L,u}}(-\cos\theta_{\text{en}} - \sin\theta_{\text{en}}), \\ r_{\text{L,u}}(\cos\theta_{\text{en}} - \sin\theta_{\text{en}}) \end{pmatrix}$
$X'_{\text{st,L,u}}$	Starting longitudinal position of the circular boundary for the upper limit while turning left	$-r_{\text{L,u}}$
$r_{\text{L,l}}$	Radius of the circular boundary for the lower limit while turning left	$[0, L_y]$
$\mathbf{c}_{\text{L,l}}$	Center of the circular boundary for the lower limit while turning left	$\begin{pmatrix} (W_{\text{en}} - r_{\text{L,l}})\sin\theta_{\text{en}} + \\ (W_{\text{ex}} - r_{\text{L,l}})\cos\theta_{\text{en}}, \\ (W_{\text{ex}} - r_{\text{L,l}})\sin\theta_{\text{en}} - \\ (W_{\text{en}} - r_{\text{L,l}})\cos\theta_{\text{en}} \end{pmatrix}$
$X'_{\text{st,L,l}}$	Starting longitudinal position of the circular boundary for the lower limit while turning left	$(W_{\text{ex}} - r_{\text{L,l}})$
$r_{\text{R,u}}$	Radius of circular boundary for the upper limit while turning right	$[0, L_y/2]$
$\mathbf{c}_{\text{R,u}}$	Center of the circular boundary for the upper limit while turning right	$\begin{pmatrix} r_{\text{R,u}}(-\cos\theta_{\text{en}} + \sin\theta_{\text{en}}), \\ -r_{\text{R,u}}(\cos\theta_{\text{en}} + \sin\theta_{\text{en}}) \end{pmatrix}$
$X'_{\text{st,R,u}}$	Starting longitudinal position of the circular boundary for the upper limit while turning right	$-r_{\text{R,u}}$
$r_{\text{R,l}}$	Radius of circular boundary for lower limit while turning right	Dependent on intersection geometry
$\mathbf{c}_{\text{R,l}}$	Center of the circular boundary for the lower limit while turning right	$\begin{pmatrix} -(W_{\text{ex}} + r_{\text{R,l}})\cos\theta_{\text{en}} + \\ (W_{\text{en}} + r_{\text{R,l}})\sin\theta_{\text{en}}, \\ -(W_{\text{ex}} + r_{\text{R,l}})\sin\theta_{\text{en}} - \\ (W_{\text{en}} + r_{\text{R,l}})\cos\theta_{\text{en}} \end{pmatrix}$
$X'_{\text{st,R,l}}$	Starting longitudinal position of the circular boundary for the lower limit while turning right	$-W_{\text{ex}} - r_{\text{R,l}}$

1) *Straight-going corridor*

The corridor for vehicles going straight is simply a rectangle connecting the entrance and exit branches.

2) *Left-turning corridor*

The corridor for vehicles turning left is delineated by smooth boundaries consisting of straight and circular parts, see red lines in Fig. 3. Specifically, the circular part of the left boundary starts at $X'_{\text{st,L,u}}$, which may be placed upstream of the crossing area to expand the turning corridor and facilitate more flexible and smoot turning. On the other hand, the circular part of the right boundary lies inside the crossing area (Fig. 3). The radii of the circular parts can be independently chosen. Based on their choice and the intersection dimensions, the centers of the circles, i.e. $\mathbf{c}_{\text{L,l}}$ and $\mathbf{c}_{\text{R,u}}$, and the beginning of the circular parts, i.e. $X'_{\text{st,L,u}}$ and $X'_{\text{st,R,u}}$, can

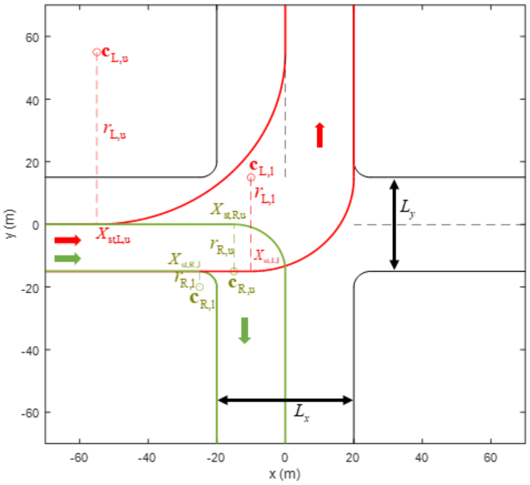

Figure 3. OD corridors for turning vehicles

be calculated, as shown in Table 1. In particular, choosing zero as the radius results in an L-shaped boundary.

3) Right-turning corridor

The right boundary of a right-turning corridor coincides with the intersection boundary, see green lines in Fig. 3. Thus, if the intersection corners are rounded, it includes a circular part; otherwise, it will be L-shaped. The left boundary comprises a straight part followed by a circular part, again with a selectable radius.

The illustrated corridors in Fig. 3 can be obtained for all other entrance branches, and the related parameters mentioned in Table 1 are valid for all ODs. The only assumption regarding this table is $L_y \le L_x \le 2L_y$, which is not too restricting. If $L_x < L_y$, we can rotate the original coordinates by 90°. Also, if $L_x > 2L_y$, we should calculate the upper limit for $r_{\mathrm{L,u}}$ such that the upper boundary for left-turning vehicles does not violate the intersection boundaries.

B. Boundary Controllers (BCs)

To ensure that vehicles remain within the designed OD corridors, two linear boundary controllers (BCs) are designed for the respective left and right boundaries, which deliver upper and lower bounds for the vehicle's steering angle. If steadily activated, a BC would guide a vehicle to the corresponding boundary asymptotically and would then have it driving on the boundary, without violating it.

Given the designed OD corridors, which are delineated by partly straight and partly circular boundaries, we can use the straight and circular BCs presented in [13]. A BC for a straight boundary with orientation θ' and lateral position y'_{d} in the transformed coordinates (Section II.B) delivers a steering angle $\delta_{\mathrm{st}}(y'(k),\theta(k),y'_{\mathrm{d}},\theta')$ that can be determined via [13]:

$$\begin{aligned}\delta_{\mathrm{st}}(y'(k),\theta(k),y'_{\mathrm{d}},\theta_{\mathrm{d}}) &= \tan^{-1}\left(\sigma\mu_{\mathrm{st}}(k)/v(k)\right)\\ \mu_{\mathrm{st}}(k) &= -\mathbf{K}_{\mathrm{st}}\begin{bmatrix} y'(k)-y'_{\mathrm{d}} \\ \theta(k)-\theta' \end{bmatrix}\end{aligned} \tag{5}$$

where $y'(k)$ and $\theta(k)$ are the vehicle's transformed lateral position and orientation, respectively, and $\mathbf{K}_{\mathrm{st}}$ is a state-feedback gain. In addition, based on [13], a BC for a circular boundary, whose radius and center are, respectively, r_{d} and $\mathbf{c}$, a steering angle $\delta_{\mathrm{cir}}(r(k),s(k),r_{\mathrm{d}},\mathbf{c})$ that reads:

$$\begin{aligned}\delta_{\mathrm{cir}}(r(k),s(k),r_{\mathrm{d}},\mathbf{c}) &= \tan^{-1}\left(\sigma\mu_{\mathrm{cir}}(k)/v(k)\right)\\ \mu_{\mathrm{cir}}(k) &= \begin{cases} -\mathbf{K}_{\mathrm{cir}}\begin{bmatrix} r(k)-r_{\mathrm{d}} \\ s(k) \end{bmatrix}+\begin{pmatrix} \dfrac{\sigma}{r_{\mathrm{d}}} \end{pmatrix} & \text{if turning left} \\ -\mathbf{K}_{\mathrm{cir}}\begin{bmatrix} r_{\mathrm{d}}-r(k) \\ s(k) \end{bmatrix}-\begin{pmatrix} \dfrac{\sigma}{r_{\mathrm{d}}} \end{pmatrix} & \text{if turning right} \end{cases}\end{aligned} \tag{6}$$

where $r(k)=\|\boldsymbol{z}(k)-\mathbf{c}\|, \boldsymbol{z}(k)=[x(k) \quad y(k)]^T$, is the Euclidean distance of current vehicle position from the center and $s(k)$ is defined as the deviation from the circular angle and is given by:

$$s(k)=\begin{cases} \theta(k)-(\varphi(k)+\pi/2) & \text{if turning left} \\ \theta(k)-(\varphi(k)-\pi/2) & \text{if turning right} \end{cases} \tag{7}$$

where $\varphi(k)=\tan^{-1}(y/x)$ is the angle of vehicle position in polar coordinates. Now, employing several parameters introduced in Table 1, we can design the boundary controllers for different intentions.

1) Straight-going boundary controller

Both lower and upper bounds, $\underline{\delta}_{\mathrm{S}}(\boldsymbol{x}(k))$ and $\overline{\delta}_{\mathrm{S}}(\boldsymbol{x}(k))$, for the steering angle can be obtained by the straight BC:

$$\begin{aligned}\overline{\delta}_{\mathrm{S}}(\boldsymbol{x}(k)) &= \delta_{\mathrm{st}}(y'(k),\theta(k),-w/2,\theta_{\mathrm{en}})\\ \underline{\delta}_{\mathrm{S}}(\boldsymbol{x}(k)) &= \delta_{\mathrm{st}}(y'(k),\theta(k),-W_{\mathrm{en}}+w/2,\theta_{\mathrm{en}})\end{aligned} \tag{8}$$

where w is the width of vehicles.

2) Left-turning boundary controller

The lower and upper bounds for the steering angle of left-turning vehicles are obtained as combinations of the straight and circular boundary parts, according to Fig. 3, as follows

$$\begin{aligned}\overline{\delta}_{\mathrm{L}}(\boldsymbol{x}(k)) &= \begin{cases} \delta_{\mathrm{st}}(y'_{\mathrm{en}}(k),\theta(k),-w/2,\theta_{\mathrm{en}}) & x'_{\mathrm{en}}(k)\le X_{\mathrm{st,L,u}} \\ \delta_{\mathrm{cir}}(r(k),s(k),r_{\mathrm{L,u}}+w/2,\boldsymbol{c}_{\mathrm{L,u}}) & \begin{matrix} x'_{\mathrm{en}}(k)>X_{\mathrm{st,L,u}}\ \& \\ y'_{\mathrm{en}}(k)<-X_{\mathrm{st,L,u}} \end{matrix} \\ \delta_{\mathrm{st}}(y'_{\mathrm{ex}}(k),\theta(k),-w/2,\theta_{\mathrm{ex}}) & \text{otherwise} \end{cases}\\ \underline{\delta}_{\mathrm{L}}(\boldsymbol{x}(k)) &= \begin{cases} \delta_{\mathrm{st}}(y'_{\mathrm{en}}(k),\theta(k),-W_{\mathrm{en}}+w/2,\theta_{\mathrm{en}}) & x'_{\mathrm{en}}(k)\le X_{\mathrm{st,L,l}} \\ \delta_{\mathrm{cir}}(r(k),s(k),r_{\mathrm{L,l}}-w/2,\boldsymbol{c}_{\mathrm{L,l}}) & \begin{matrix} x'_{\mathrm{en}}(k)>X_{\mathrm{st,L,l}}\ \& \\ y'_{\mathrm{en}}(k)<r_{\mathrm{L,l}}-W_{\mathrm{en}} \end{matrix} \\ \delta_{\mathrm{st}}(y'_{\mathrm{ex}}(k),\theta(k),-W_{\mathrm{ex}}+w/2,\theta_{\mathrm{ex}}) & \text{otherwise} \end{cases}\end{aligned} \tag{9}$$

where $(.)'_{\mathrm{en}}$ and $(.)'_{\mathrm{ex}}$ represent the transformed variables in the coordinates aligned with the entrance and exit branches, respectively.

3) Right-turning boundary controller

Similarly, the lower and upper bounds for the steering angle of right-turning vehicles are described by:

$$\begin{aligned}\overline{\delta}_{\mathrm{R}}(\boldsymbol{x}(k)) &= \begin{cases} \delta_{\mathrm{st}}(y'_{\mathrm{en}}(k),\theta(k),-w/2,\theta_{\mathrm{en}}) & x'_{\mathrm{en}}(k)\le X_{\mathrm{st,R,u}} \\ \delta_{\mathrm{cir}}(r(k),s(k),r_{\mathrm{R,u}}-w/2,\boldsymbol{c}_{\mathrm{R,u}}) & \begin{matrix} x'_{\mathrm{en}}(k)>X_{\mathrm{st,R,u}}\ \& \\ y'_{\mathrm{en}}(k)>-r_{\mathrm{R,u}} \end{matrix} \\ \delta_{\mathrm{st}}(y'_{\mathrm{en}}(k),\theta(k),-w/2,\theta_{\mathrm{ex}}) & \text{otherwise} \end{cases}\\ \underline{\delta}_{\mathrm{R}}(\boldsymbol{x}(k)) &= \begin{cases} \delta_{\mathrm{st}}(y'_{\mathrm{en}}(k),\theta(k),-W_{\mathrm{en}}+w/2,\theta_{\mathrm{en}}) & x'_{\mathrm{en}}(k)\le X_{\mathrm{st,R,l}} \\ \delta_{\mathrm{cir}}(r(k),s(k),r_{\mathrm{R,l}}+w/2,\boldsymbol{c}_{\mathrm{R,l}}) & \begin{matrix} x'_{\mathrm{en}}(k)>X_{\mathrm{st,R,l}} \\ \& y'_{\mathrm{en}}(k)>-W_{\mathrm{en}} \end{matrix} \\ \delta_{\mathrm{st}}(y'_{\mathrm{ex}}(k),\theta(k),-W_{\mathrm{ex}}+w/2,\theta_{\mathrm{ex}}) & \text{otherwise} \end{cases}\end{aligned} \tag{10}$$

C. Desired Orientations

Depending on the vehicle's intention and current position, a desired orientation $\theta_{\mathrm{d},i}(\boldsymbol{x}(k))$ is generated that guides the vehicle smoothly towards its destination; however, vehicles may deviate from the ideal path to meet more important goals, like avoiding collision. For the vehicles going straight, the desired orientation is constant, i.e. $\theta_{\mathrm{d}}(\boldsymbol{x}(k)) = \theta_{\mathrm{en}} = \theta_{\mathrm{ex}}$. For turning vehicles, the general idea is to gradually change the desired orientation from the entrance branch to the exit branch orientation. To this end, we define a turning interval for the transformed longitudinal position, denoted $[x_{\mathrm{start}}, x_{\mathrm{end}}]$, in which the desired orientation changes linearly from the entrance branch to the exit branch orientation. Before and after this interval, the desired orientation equals the entrance and exit branch orientation, respectively. Thus, the desired orientation is calculated by:

$$\theta_{\mathrm{d}}(\boldsymbol{x}(k)) = \begin{cases} \theta_{\mathrm{en}} & x'_{\mathrm{en}}(k) \le x_{\mathrm{start}} \\ \dfrac{\theta_{\mathrm{ex}} - \theta_{\mathrm{en}}}{x_{\mathrm{end}} - x_{\mathrm{start}}}\left(x'_{\mathrm{en}}(k) - x_{\mathrm{start}}\right) + \theta_{\mathrm{en}} & x_{\mathrm{start}} < x'_{\mathrm{en}}(k) \le x_{\mathrm{end}} \\ \theta_{\mathrm{ex}} & x'_{\mathrm{en}}(k) > x_{\mathrm{end}} \end{cases} \tag{11}$$

where $x'_{\mathrm{en}}(k)$ is the vehicle longitudinal position in the rotated coordinates aligned with the entrance branch. In this context, it is reasonable to properly distribute vehicles in the exit branch so as to avoid crossing trajectories of vehicles with the same OD and the resulting unnecessary conflicts. To this end, x_{end} is determined according to the vehicle's initial lateral position. Specifically, if a vehicle starts on the left (right) boundary of the entrance, it will end up on the left (right) boundary of the exit branch. For any other value in-between, a proportional value is found by:

$$x_{\mathrm{end}} = \begin{cases} -W_{\mathrm{ex}} y'_{\mathrm{en}}(0) / W_{\mathrm{en}} & \text{if turning left} \\ W_{\mathrm{ex}} y'_{\mathrm{en}}(0) / W_{\mathrm{en}} & \text{if turning right} \end{cases}. \tag{12}$$

Note that $y'_{\mathrm{en}}(0)$, the initial transformed lateral position, ranges from $-w/2$ (left boundary) to $-W_{\mathrm{en}} + w/2$ (right boundary). As a result, x_{end} has a positive (negative) value for left-turning (right-turning) vehicles. Since there is no obstacle for turning left, a constant x_{start} can be taken for all left-turning vehicles. On the other hand, if a right-turning vehicle is close to the right boundary, it cannot start turning before reaching the intersection corner because it would touch the boundary. Therefore, a lateral position-dependent x_{start} is determined for right-turning vehicles. Thus, the following equation determines x_{start}

$$x_{\mathrm{start}} = \begin{cases} -L_{\mathrm{start,L}} - (W_{\mathrm{ex}} + r_{\mathrm{R,1}}) & \text{if turning left} \\ -\dfrac{L_{\mathrm{start,R}}}{W_{\mathrm{ex}}}(x_{\mathrm{end}} + W_{\mathrm{ex}} + r_{\mathrm{R,1}}) - (W_{\mathrm{ex}} + r_{\mathrm{R,1}}) & \text{if turning right} \end{cases} \tag{13}$$

where $L_{\mathrm{start,L}}$ and $L_{\mathrm{start,R}}$ are the maximum distance from the intersection at which left-turning and right-turning vehicles may start turning and $r_{\mathrm{R,1}}$ is the radius of the intersection corners, see. Fig. 4. Specifically, if $L_{\mathrm{start,L}} = L_{\mathrm{start,R}} = 0$, all vehicles start their turning interval once reaching the crossing area, i.e. $x'_{\mathrm{en}} = -W_{\mathrm{ex}} - r_{\mathrm{R,1}}$.

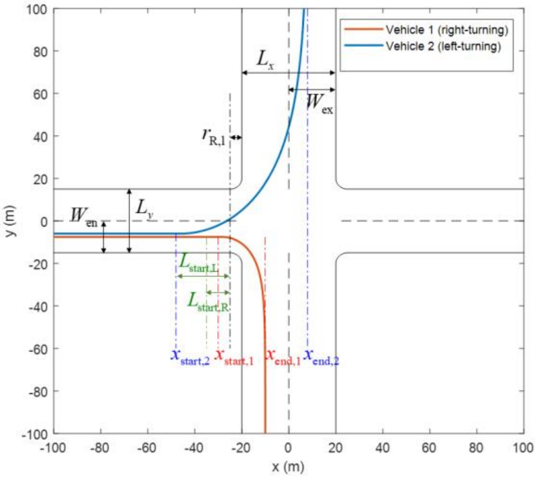

Figure 4. The turning interval

The proposed strategy does not force vehicles to end up in a pre-defined position. Instead, a desired lateral position in the exit branch is defined to be pursued, if possible. In the case of exceeding x_{end}, while still in the intersection, a vehicle will not return to reach its desired lateral position, as evidenced from the third line of (11). This flexibility may prevent unnecessary turning on the intersection. Fig. 4 illustrates samples of movements of two turning vehicles, crossing alone the intersection, hence following the ideal path determined by the sequence of desired orientations.

IV. Optimal Control Problem

In this section, we formulate a joint optimal control problem that minimizes a weighted sum of multiple sub-objectives for all vehicles to ensure safe, smooth, and efficient movements. The cost function, constraints, and numerical solution algorithm are presented.

A. Objective Function

We address several goals, which are generally similar to the ones presented in [6], but are modified here for the bicycle model and intersection context. The goals are reflected in corresponding weighted terms of the cost function as described below.

1) Fuel consumption and passenger comfort

As demonstrated in [28], fuel-minimizing vehicle trajectories in OCPs are obtained when minimizing the square-of-acceleration. Therefore, we include quadratic terms of accelerations $F_i^2(k)$, but also of steering angles $\delta_i^2(k)$, in the cost function, as minimizing the accelerations and steering angles also accounts for passenger comfort.

2) Desired speed tracking

Each crossing vehicle i has a desired speed $v_{\mathrm{d},i}$. Although accurately tracking the desired speed is not the most important goal, including a term $(v_i(k) - v_{\mathrm{d},i})^2$ in the objective function penalizes deviations from the desired speed, thus influencing vehicle advancement and intersection throughput.

3) Desired orientation tracking

To drive close to the ideal path, whenever possible, a term penalizing deviation from the desired orientation

$(\theta_i(k)-\theta_{\mathrm{d},i}(\boldsymbol{x}(k)))^2$ is included in the cost function where $\theta_{\mathrm{d},i}(\boldsymbol{x}(k))$ is the desired orientation determined by (11).

4) Collision avoidance

Given the serious potential conflicts among vehicles with antagonistic ODs, it is important to avoid collisions. To meet this objective, we define an elliptic distance between each pair of vehicles:

$$d_{i,j}(k)=\sqrt{\left(x_i'(k)-x_j'(k)\right)^2+p\left(y_i'(k)-y_j'(k)\right)^2} \qquad (14)$$

where $x'(k)$ and $y'(k)$, for both vehicles, are the longitudinal and lateral positions in the coordinates aligned with vehicle i orientation, and are calculated by (2) with $\theta'=\theta_i(k)$. Also, $p\ge 1$ is a parameter determining the ellipsoid shape, i.e. its length versus width. Given the vehicles' rectangular shape and forward speed, the ellipsoid is elongated in its movement direction, without wasting space in the perpendicular direction. Then, a decreasing function is applied to the introduced distance:

$$c_{i,j}(k)=\gamma_1\exp\left(-d_{i,j}(k)/\gamma_2\right) \qquad (15)$$

where γ_1 and γ_2 are constant parameters that tune the intensity and magnitude of the surrounding aura. Note that, if orientations of two vehicles are different, $c_{i,j}(k)$ and $c_{j,i}(k)$ are not necessarily equal; hence, both terms must be included in the cost function. Fig. 5 depicts the elliptic iso-cost curves of the collision avoidance function.

Considering the mentioned sub-objectives, the cost function is:

$$J=\sum_{k=0}^{K-1}\sum_{i=1}^{n}[\{\omega_1F_i^2(k)+\omega_2\delta_i^2(k)+\omega_3(v_i(k)-v_{d,i})^2 \\ +\omega_4(\theta_i(k)-\theta_{\mathrm{d},i}(\boldsymbol{x}(k)))^2+\omega_5\sum_{\substack{j=1\\ j\ne i}}^{n}c_{i,j}(k)\}] \qquad (16)$$

where K is the fixed time horizon which should be chosen sufficiently long so that vehicles have enough time to cross the intersection. Also, n is the number of vehicles and $\omega_i, i=1,2,...,5$, are the positive weights that determine the relative importance of the included terms. No fixed final states are considered for the vehicles, which may improve the intersection throughout because vehicles are flexibly advancing and do not maneuver or decelerate to match pre-defined final positions.

B. Control Input Constraints

To reflect physical limitations, guarantee boundary respect, and foster passenger comfort, some control input constraints are considered.

1) Acceleration constraints

The first issue for acceleration is practical limits, as each vehicle has a range of feasible accelerations, $\left[F_{\min},F_{\max}\right]$. Moreover, the acceleration should be restricted to a lower bound to avoid negative speed. Given the state equation of speed, we finally construct the acceleration constraints as:

$$\begin{aligned}&\underline{F}(\boldsymbol{x}(k))\le F(k)\le\bar{F}(\boldsymbol{x}(k))\\&\underline{F}(\boldsymbol{x}(k))=\max(F_{\min},-v(k)/T)\\&\bar{F}(\boldsymbol{x}(k))=F_{\max}\end{aligned} \qquad (17)$$

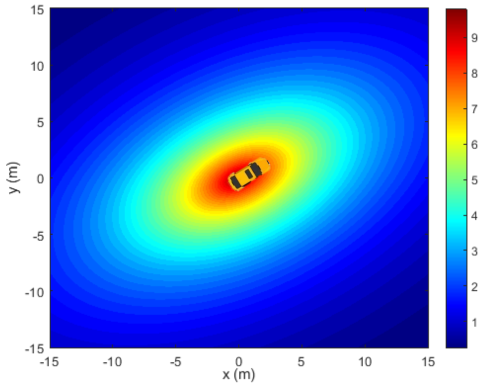

Figure 5. Collision avoidance function iso-cost curves

2) Steering angle constraints

Two sets of bounds are considered for the steering angle, the first of which are the OD boundary controllers presented in Section III.B. Besides, the centrifugal acceleration must not exceed a comfort threshold. A vehicle with constant steering value drives a circle with radius

$$r=\sigma/\left|\tan(\delta(k))\right| \qquad (18)$$

and centrifugal acceleration

$$F_{\mathrm{c}}=v^2\left|\tan(\delta(k))\right|/\sigma \qquad (19)$$

which should remain less than a pre-defined threshold $F_{\mathrm{c,max}}$. Hence, the steering angle should satisfy the following inequality:

$$-\tan^{-1}(\sigma F_{\mathrm{c,max}}/v^2(k))\le\delta(k)\le\tan^{-1}(\sigma F_{\mathrm{c,max}}/v^2(k)) . \qquad (20)$$

Finally, the two mentioned sets are combined. For instance, the constraints for a straight-going vehicle is

$$\begin{aligned}&\underline{\delta}(\boldsymbol{x}(k))\le\delta(k)\le\bar{\delta}(\boldsymbol{x}(k))\\&\underline{\delta}(\boldsymbol{x}(k))=\max\left(-\tan^{-1}(\sigma F_{\mathrm{c,max}}/v^2(k)),\underline{\delta}_{\mathrm{S}}(\boldsymbol{x}(k))\right)\\&\bar{\delta}(\boldsymbol{x}(k))=\min\left(\tan^{-1}(\sigma F_{\mathrm{c,max}}/v^2(k)),\bar{\delta}_{\mathrm{S}}(\boldsymbol{x}(k))\right)\end{aligned} \qquad (21)$$

It should be mentioned that in (17) and (21), smooth maximum and minimum functions are involved. Note also that, in this work, the state-dependent control bounds are transformed to constant control bounds by replacing the original control variables F_i and δ_i as follows (see also [6], [26]):

$$F_i(k)=(1-u_{F,i}(k))\underline{F}_i(\boldsymbol{x}(k))+u_{F,i}(k)\bar{F}_i(\boldsymbol{x}(k)) \qquad (22)$$

$$\delta_i(k)=(1-u_{\delta,i}(k))\underline{\delta}_i(\boldsymbol{x}(k))+u_{\delta,i}(k)\bar{\delta}_i(\boldsymbol{x}(k)) \qquad (23)$$

where $u_{F,i}$ and $u_{\delta,i}$ are the new control variables, with constant bounds $0\le u_{F,i},u_{\delta,i}\le 1$, for each vehicle i.

C. Problem Formulation

Summarizing, the general form of the OCP reads

$$\text{Minimize } J=\sum_{k=0}^{K-1}\Phi[\boldsymbol{x}(k),\boldsymbol{u}(k)] \qquad (24)$$

$$\boldsymbol{x}(k+1)=\boldsymbol{f}[\boldsymbol{x}(k),\boldsymbol{u}(k)] \qquad (25)$$

$$\boldsymbol{u}^{\min}\le\boldsymbol{u}(k)\le\boldsymbol{u}^{\max} \qquad (26)$$

where $\boldsymbol{u}^{\min}$ and $\boldsymbol{u}^{\max}$ are the constant lower and upper control bounds. The Hamiltonian function is described as follows (see [26], [29]):

$$H[(\boldsymbol{x}(k),\boldsymbol{u}(k),\boldsymbol{\lambda}(k+1)] = \Phi(\boldsymbol{x}(k),\boldsymbol{u}(k)) + \boldsymbol{\lambda}(k+1) f[\boldsymbol{x}(k),\boldsymbol{u}(k)] \quad (27)$$

where $\boldsymbol{\lambda}(k)$ is the co-states vector. The following necessary conditions of optimality, need to be satisfied for $k = 0,...,K-1$

$$\boldsymbol{x}(k+1) = \partial H / \partial \boldsymbol{\lambda}(k+1) = \boldsymbol{f}[\boldsymbol{x}(k),\boldsymbol{u}(k)] \quad (28)$$

$$\boldsymbol{\lambda}(k) = \partial H / \partial \boldsymbol{x}(k) \quad (29)$$

$$\frac{\partial H}{\partial u_i(k)} \begin{cases} < 0 & \text{if } u_i(k) = u_i^{\max} \\ = 0 & \text{if } u_i^{\min} \le u_i(k) \le u_i^{\max} \\ > 0 & \text{if } u_i(k) = u_i^{\min} \end{cases} \quad (30)$$

with (28) and (29) being the state and co-state difference equations, and (30) specifying the optimal control variables u_i . Finally, the boundary conditions are given by:

$$\boldsymbol{x}(0) = \boldsymbol{x}_0 \quad (31)$$

$$\boldsymbol{\lambda}(0) = 0 \quad (32)$$

D. Numerical Solution

The formulated OCP is solved with the use of a very efficient algorithm, i.e., FDA [6],[26]. The algorithm exploits the structure of the state equations, mapping the OCP into a Nonlinear Programming problem in the reduced space of control variables, i.e., in an mK-dimensional space, where m is the number of control variables. The algorithm yields a control trajectory $\boldsymbol{u}(k)$, $k = 0,...,K-1$, which corresponds to a local minimum of the objective function, while satisfying the state equations and the constraints. To this end, FDA exploits the fact that $\boldsymbol{g}(k) = \left[\partial \boldsymbol{f} / \partial \boldsymbol{u}(k)\right]^T \boldsymbol{\lambda}(k+1) + \partial \Phi / \partial \boldsymbol{u}(k)$ is the reduced gradient in the mK-dimensional reduced space of the control, if the states and co-states involved in the partial derivatives satisfy the state and co-state equations.

FDA is an iterative algorithm which starts with a given feasible initial control trajectory. Each iteration attempts to improve the control trajectories, by calculating an appropriate step in the mK-dimensional control space. In this work, the step calculation is derived based on the resilient backpropagation (RPROP) method, for more details see [6], [30]-[31].

V. Simulation Results

In order to test the effectiveness of the proposed strategy, several intersection crossing scenarios were considered, and one of them, involving 21 vehicles, is reported here for demonstration. The initial conditions of these vehicles are chosen according to the second (and more challenging) scenario of [23]. Length and width of vehicles are, respectively, $\sigma = 4.5$ m and $w = 1.7$ m ; the dimension of the intersection is $L_x = L_y = 30$ m; OD corridor parameters are $L_{\text{start,L}} = 15$ m, $L_{\text{start,R}} = 10$ m, $r_{\text{L,l}} = 30$ m, $r_{\text{L,u}} = 45$ m, $r_{\text{R,l}} = 5$ m, $r_{\text{R,l}} = 5$ m, and $r_{\text{R,u}} = 15$ m. Also, $K = 100$, $T = 0.1$ s, $p = 7$, $F_{\text{c,max}} = 5$ m/s^2, $F_{\min} = -3$ m/s^2, $F_{\max} = 1.5$ m/s^2. Finally, the cost function weights are $\omega = [0.005, 1, 0.002, 0.5, 1.5]$. Each vehicle can have its own desired speed; however, in this work, we consider the same desired speed for all vehicles with the same intentions. Specifically, 12 m/s, 10 m/s, and 8 m/s are respectively considered for straight, left-turning, and right turning vehicles. FDA is developed in C++ environment and run on a PC with an Intel(R) Core™i5-10500 CPU @ 3.10GHz with 8.0 GB of installed RAM. The optimal results are illustrated in Fig. 6-7. As observed in Fig. 6, vehicles smoothly move towards their destination while remaining within the intersection boundaries as well as their respective OD corridors. Vehicles may deviate from their desired orientation to avoid collisions. The initial speed of each vehicle is 10 m/s and after a while, they reach the defined desired speed, depending on their intention. Note that collision avoidance actions may make them also deviate from their desired speeds, as shown in Fig. 7. The vehicle control inputs remain in the admissible range. Due to the fact that the vehicles' initial positions are relatively close, bearing many conflicts, appropriate accelerations, decelerations, and steering angles are generated to provide a collision-free solution. With easier initial conditions, the fluctuations in the control signals are much less pronounced. These severe initial conditions require 200 iterations to reach a collision-free solution with sufficiently smooth trajectories that take 3.46 s while the reported computational time for the same crossing scenario in [23] is 89 s. A video of the results can be seen at https://bit.ly/3Tuxvwg.

The needed computation time is promising for the next steps of the research in which we will implement a real-time MPC approach. Note that in the MPC scheme, the problem is solved repeatedly, hence a shorter horizon may be used; also in addition, using the previous solution to construct the starting trajectories for FDA leads to less iterations. Indeed, first results indicate that the computation of the online solution takes around one model sampling period (0.2 s).

VI. Conclusions

A joint optimal control approach is developed for vehicles crossing signal-free intersections which are inherently lane-free. A cost function is defined to reflect our major goals, including minimizing control efforts, avoiding collisions, moving smoothly toward the destination, and following the desired speed, when possible. Additionally, control input constraints are formed to guarantee remaining within the intersection, satisfying practical limits, and providing passenger comfort. The described problem is then solved by a fast and efficient numerical algorithm. Results verify that the proposed method is effective and relatively fast. Compared with the existing works, the proposed approach significantly reduces the computational burden and releases some limiting assumptions like pre-defined terminal position or fixed exit time for all vehicles.

References

[1] M. Papageorgiou, C. Diakaki, V. Dinopoulou, A. Kotsialos, and Y. Wang, "Review of road traffic control strategies", *Proceedings of the IEEE* ,91, 2003, pp. 2043-2067.

[2] C. Diakaki, M. Papageorgiou, I. Papamichail, and I. Nikolos, "Overview and analysis of vehicle automation and communication systems from a motorway traffic management perspective", *Transportation Research Part A*, 75, 2015, pp. 147–165.

[3] I. Papamichail, N. Bekiaris-Liberis, A.I. Delis, D. Manolis, K.-S. Mountakis, I.K. Nikolos, C. Roncoli, and M. Papageorgiou, "Motorway traffic flow modelling, estimation and control with vehicle automation and communication systems", *Annual Reviews in Control*, 48 ,2019, pp. 325-346.

[4] M. Papageorgiou, K.S. Mountakis, I. Karafyllis, I. Papamichail, and Y. Wang, “Lane-free artificial-fluid concept for vehicular traffic”, *Proceedings of the IEEE*, 109, 2021, pp. 114-121.

[5] M. Malekzadeh, D. Manolis, I. Papamichail and M. Papageorgiou, "Empirical Investigation of Properties of Lane-free Automated Vehicle Traffic", *IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*, Macau, China, 2022, pp. 2393-2400.

[6] V.K. Yanumula, P. Typaldos, D. Troullinos, M. Malekzadeh, I. Papamichail, and M. Papageorgiou, “Optimal trajectory planning for connected and automated vehicles in lane-free traffic with vehicle nudging”, *IEEE Transaction on Intelligent Vehicles*, 8(3), 2023, pp. 2385-2399.

[7] N. Dabestani, P. Typaldos, V. K. Yanumula, I. Papamichail and M. Papageorgiou, "Joint Trajectory Optimization for Multiple Automated Vehicles in Lane-free Traffic with Vehicle Nudging," *IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*, Bilbao, Spain, 2023, pp. 5954-5961.

[8] D. Troullinos, G. Chalkiadakis, I. Papamichail, and M. Papageorgiou, “Collaborative multiagent decision making for lane-free autonomous driving”, *20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, online, 2021, pp. 1335-1343.

[9] I. Karafyllis, D. Theodosis, and M. Papageorgiou, “Lyapunov-based two-dimensional cruise control of autonomous vehicles on lane-free roads", *Automatica*, 145, 2022, 110517.

[10] M. Sekeran, M. Rostami-Shahrbabaki, A.A. Syed, M. Margreiter, and K. Bogenberger, “Lane-free traffic: History and state of the art”, *IEEE 25th IEEE International Intelligent Transportation Conference (ITSC)*, Macao, China, 2022, pp. 1037-1042.

[11] M. Naderi, M. Papageorgiou, I. Karafyllis and I. Papamichail, "Automated vehicle driving on large lane-free roundabouts," *IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*, Macau, China, 2022, pp. 1528-1535

[12] M. Naderi, M. -K. Mavroeidi, I. Papamichail and M. Papageorgiou, "Optimal Orientation for Automated Vehicles on Large Lane-Free Roundabouts," *62nd IEEE Conference on Decision and Control (CDC)*, Singapore, 2023, pp. 8207-8214.

[13] M. Naderi, M. Papageorgiou, D. Troullinos, I. Karafyllis and I. Papamichail, "Controlling Automated Vehicles on Large Lane-Free Roundabouts," IEEE Transactions on Intelligent Vehicles, 9(1), 2024, pp. 3061-3074.

[14] M. Papageorgiou, M. Ben-Akiva, J. Bottom, P.H.L. Bovy, S.P. Hoogendoorn, N.B. Hounsell, A. Kotsialos, M. McDonald, “ITS and Traffic Management”, *Handbooks in operations research and management Science*, 14, C. Barnhart and G. Laporte, Editors, North-Holland (Elsevier), 2007, pp. 715-774.

[15] L. Chen and C. Englund, “Cooperative intersection management: A survey”, *IEEE Transactions on Intelligent Transportation Systems*, 17, 2016, pp. 570-586.

[16] E. Namazi, J. Li, and C. Lu, "Intelligent intersection management systems considering autonomous vehicles: A systematic literature review", *IEEE Access*, 7 , 2019, pp. 91946-91965.

[17] J. Wu and X. Qu, “Intersection control with connected and automated vehicles: a review”, *Journal of Intelligent and Connected Vehicles*, 5(3), 2022, pp. 260-269.

[18] B. Li and Y. Zhang, "Fault-tolerant cooperative motion planning of connected and automated vehicles at a signal-free and lane-free intersection", *IFAC-PapersOnLine*, 51(24), 2018, pp. 60-67.

[19] B. Li, Y. Zhang, N. Jia, and X. Peng, “Autonomous intersection management over continuous Space: A microscopic and precise solution via computational optimal control”, *IFAC-PapersOnLine*, 53(2), 2020, pp. 17071-17076.

[20] B. Li, Y. Zhang, T. Acarman, Y. Ouyang, C. Yaman, and Y. Wang, “Lane-free autonomous intersection management: A batch processing framework integrating reservation-based and planning-based methods”, *IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 7915-7921.

[21] B. Li, D. Cao, S. Tang, T. Zhang, H. Dong, Y. Wang, and F.Y. Wang, “Sharing traffic priorities via cyber–physical–social intelligence: A lane-free autonomous intersection management method in metaverse”. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 53(4), 2023, pp. 2025-2036.

[22] M. Amouzadi, M.O. Orisatoki, and A.M. Dizqah, “Lane-free crossing of CAVs through intersections as a minimum-time optimal control problem”. *IFAC-PapersOnLine*, 55(14), 2022, pp.28-33.

[23] M. Amouzadi, M. O. Orisatoki, and A. M. Dizqah, "Optimal lane-free crossing of CAVs through intersections,", *IEEE Transactions on Vehicular Technology*, 72(2), 2022, pp. 1488-1500.

[24] E. Ahmadi, and R.C. Carlson, “Model predictive control of connected vehicles under automated driving at path-free signal-free intersections”, *Transportes*, 31(3), 2022, pp. 1-16.

[25] R.C. Carlson, E. Ahmadi, E.R. Müller, and G.W.S. Silva, “Optimal coordination of connected vehicles under automated driving at path-free signal-free urban intersections”, *36th Congresso de Pesquisa Ensino em Transportes*, Fortaleza, Brazil, 2022.

[26] M. Papageorgiou, M. Marinaki, P. Typaldos, and K. Makantasis, “A feasible direction algorithm for the numerical solution of optimal control problems–extended version,” Chania, Greece: Technical University of Crete, Dynamic Systems and Simulations Laboratory, 2016.

[27] D. Theodosis, F.N. Tzortzoglou, I. Karafyllis, I. Papamichail, and M. Papageorgiou, “Sampled-data controllers for autonomous vehicles on lane-free roads”, *30th Mediterranean Conference on Control and Automation (MED)*, Athens, Greece, 2022, pp. 103-108.

[28] P. Typaldos, I. Papamichail, M. Papageorgiou, “Minimization of fuel consumption for vehicle trajectories”, *IEEE Transaction on Intelligent Transportation Systems*, 21, 2020, pp. 1716-1726.

[29] M. Papageorgiou, M. Leibold, and M. Buss, “*Optimierung”*, 4, Springer Berlin Heidelberg, 2015.

[30] A. Kotsialos and M. Papageorgiou, "Nonlinear optimal control applied to coordinated ramp metering," *IEEE Transactions on Control Systems Technology*, 12(6), 2004, pp. 920-933.

[31] A. Kotsialos, "Nonlinear optimization using directional step lengths based on RPROP," *Optimization Letters*, 8, pp. 1401-1415, 2014.

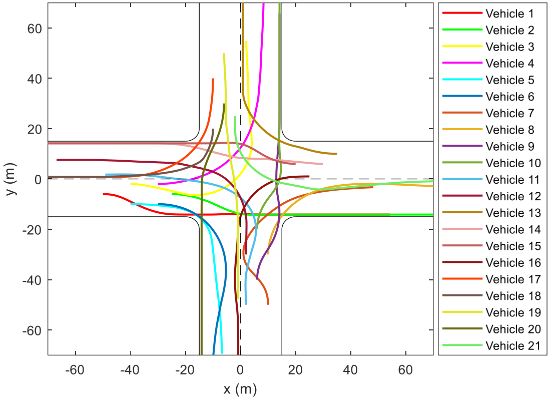

Figure 6. Vehicle paths

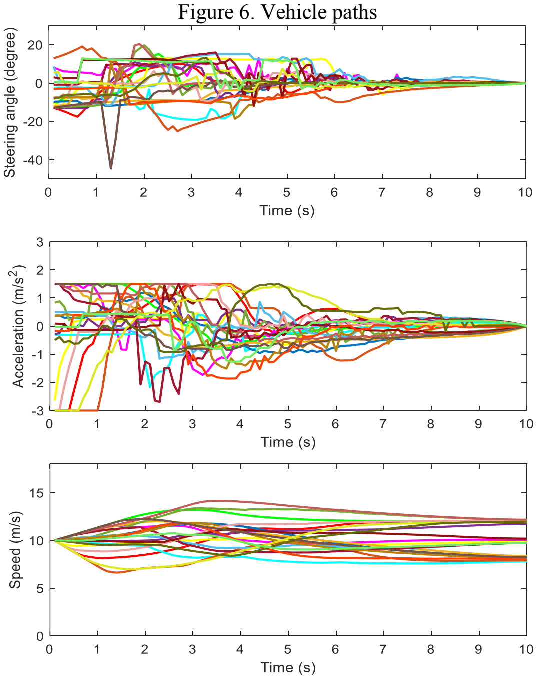

Figure 7. Vehicle steering angles, accelerations, and speeds