

Near-Optimal Coordination of Vehicles at an Intersection Plaza Using Bézier Curves

Elham Ahmadi
 Rodrigo Castelan Carlson
 Werner Kraus Junior
Federal University of Santa Catarina
 Florianópolis, SC, Brazil
 email: elham.ahmadi@posgrad.ufsc.br
 email: {rodrigo.carlson,werner.kraus}@ufsc.br

Ehsan Taheri
Department of Aerospace Engineering
Auburn University
 Auburn, AL, USA
 email: etaheri@auburn.edu

Abstract—We investigate the problem of optimal coordination of Connected Vehicles under Automated Driving (CVAD) at intersections. We aim for more driving flexibility for the CVAD, with the possibility of fully utilizing the intersection space, while strictly avoiding collisions. We propose the Intersection Trajectories Optimal control Problem (ITOP), in which an intersection is a space without movement-related horizontal markings or structural restrictions, except for the intersection boundaries, which we call a *plaza*. By using the Bézier curves and discretization, we convert the ITOP to a Non-Linear Program (NLP) that generates near-optimal collision-free trajectories. Numerical results demonstrate that the proposed approach generates feasible trajectories and is suitable for the solution of the ITOP.

Index Terms—*intersection Plaza; Bézier curves; optimal control; non-linear programming*.

I. INTRODUCTION

In spite of the changes taking place in traffic systems due to the emergence of Connected Vehicles under Automated Driving (CVAD), the paradigm established decades ago for the organization of road infrastructure remains roughly the same. Most of the vehicle coordination strategies and intersection models proposed in this new context [1] [2] rely on the concept of vehicle movements, thus restricting the possible or allowed paths within the intersection. This configures a waste of scarce intersection space and a loss of efficiency.

The proposed approaches for modeling intersections can be summarized in three categories [1] [3]:

- Cells: the intersection is divided into cells while time windows in each cell are allocated to vehicles so that there are no collisions (Fig. 1a). See, e.g., [4] [5].
- Paths: a limited set of paths are allocated to vehicles in such a way that vehicles on conflicting paths do not collide (Fig. 1b). See, e.g., [6].
- Conflicting regions/points: only the points or regions where conflicts between paths occur are discretized and time windows are allocated to vehicles to pass through these points (Fig. 1c). See, e.g., [7]–[9].

Moreover, most of these strategies simplify the behavior within the intersection to consider a constant speed and some even preclude turning movements.

In essence, these models deal with how to allocate time windows of the scarce intersection space to different vehicles. However, despite the higher capacity obtained by the elimination of the traffic light cycle and by the smaller headways between vehicles enabled by CVAD, the capacity of the intersection ends up limited by the relationship between paths constrained to pre-established vehicular movements.

Better use of the infrastructure is possible if the vehicles are allowed to make full use of the intersection space via the definition of their trajectories for any possible path. The research on this subject is limited. In [10], a cooperative motion-planning method was proposed for Connected and Automated Vehicles (CAVs) crossing an intersection. The solution of a centralized optimal control problem provides optimized trajectories offline based on pre-defined formations to which the CAVs are guided online before entering the intersection. The high-dimensionality and non-linearity of the model called for the convexification of the collision-avoidance constraints and an algorithm that provides suitable trajectories as initial guesses for the problem to speed up the solution [11]. An approach relying on emergent behaviors modeled vehicle behavior by a hierarchical set of rules, similar to the modeling of flocks [12] [13]. The simplicity of the approach results in low computation complexity but also in high sensitivity to minor changes in the rules. Finally, [14] proposed formal methods to model the behavior at intersections. The nature of the formulation and the employed methods lead, however, to combinatorial a explosion of the states even for small

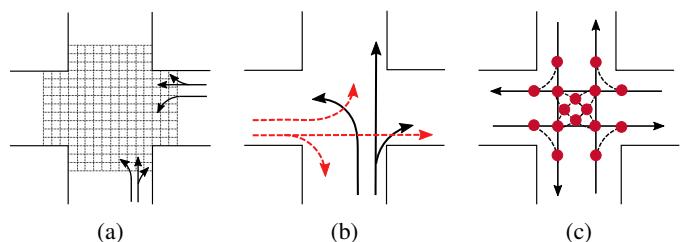


Fig. 1: Intersection modeling: (a) cells; (b) paths; and (c) collision regions/points (red circles) (adapted from [1]).

instances.

We consider an intersection as an empty space free of movement-related horizontal markings or structural restrictions, except for its boundaries. We call this space a *plaza* (see Fig. 2). The objective is to find the optimal trajectories, without predefined paths, that vehicles should follow to minimize one or more criteria, such as intersection delay or fuel consumption, while avoiding collisions. To this end, we state an Intersection Trajectories Optimal control Problem (ITOP) for CVAD. To solve the ITOP, we propose finding general functions to describe the optimal trajectories of the vehicles according to their positions and speeds, and plaza geometry.

In a previous work [15], we explored Finite Fourier Series (FFS) for the generation of trajectories (see also [16]). In this work, Bézier curves [17]–[19] are used along with discretization notions to convert the ITOP into a Non-Linear Program (NLP) with Bézier coefficients as the unknown parameters. This paper is featured by: (i) the vehicles' paths and speeds are no longer fixed; and the intersection is signal-free and movement-free. The main contributions of this paper are:

- The effective collision avoidance constraints;
- A method for the generation of the near-optimal trajectories that utilizes Bézier curves; and
- A compact representation of the Bézier curve that reduces the number of decision variables.

In Section II, the model details are provided and the ITOP is stated. In Section III, the ITOP is formulated as a NLP based on the Bézier curves representation of the states and discretization of the problem. Numerical results are presented in Section IV. Section V concludes the paper.

II. OPTIMAL CONTROL PROBLEM

In this section, the intersection plaza is modeled, the vehicles' state equations, the constraints, and the performance criterion are presented, and the ITOP is stated.

A. Plaza Modeling

A plaza can have varied layouts with respect to the number of intersecting roads and geometry. To introduce our concept, we select a four-leg intersection (Fig. 2). The x and y axes represent the central lines of the plaza on the Cartesian Coordinate System (CCS). Vehicles can travel between any two points of approaching and departing roads and their trajectories, e.g., T_1 and T_2 in the figure, are not bound to pre-specified paths or movements. We model this plaza by its four Intersection Boundaries (IB) shown by the dashed lines in Fig. 2. Each IB_h , $h = 1, \dots, 4$, is modeled by an exponential function:

$$\begin{aligned} y_h &= f_h(x(t)), \\ f_h(x(t)) &= r_{0,h} + r_{1,h} \cdot e^{r_{2,h} \cdot (x(t) + r_{3,h})}, \end{aligned} \quad (1)$$

with parameters $r_{0,h}$, $r_{1,h}$, $r_{2,h}$, and $r_{3,h}$ that shape the function according to the intersection geometry.

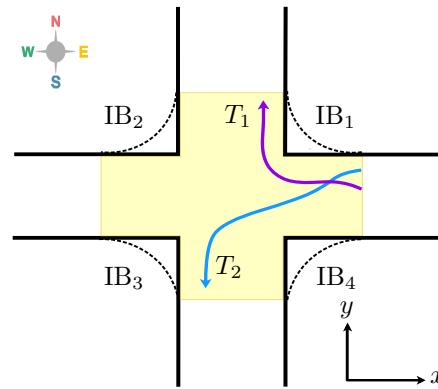


Fig. 2: A typical four-leg intersection as a plaza (colored area) with approximated boundaries (dashed lines).

B. Vehicle's Equations of Motion

There are various models to describe the vehicle's dynamics, from the simple unicycle model [20] to sophisticated car models [21]. For simplicity, in this work we use Equations of Motion (EoM) that model vehicles as point masses:

$$\begin{cases} \ddot{x}_j(t) = a_{x_j}(t), \\ \ddot{y}_j(t) = a_{y_j}(t), \end{cases} \quad (2)$$

with a_{x_j} and a_{y_j} the acceleration of vehicle j in coordinates x and y in the CCS, respectively, and t the continuous time. The total (absolute) acceleration of vehicle j is given by:

$$a_j(t) = \sqrt{a_{x_j}^2(t) + a_{y_j}^2(t)}. \quad (3)$$

The speed increment of the j -th vehicle is defined as:

$$\Delta v_j = \int_0^T a_j(t) dt, \quad (4)$$

in which T stands for the completion time, i.e., the time taken by the vehicles to cross the plaza. Then, given k vehicles at the plaza, the total speed increment is computed as:

$$\Delta v = \Delta v_1 + \dots + \Delta v_k. \quad (5)$$

C. Constraints

1) *Vehicle's kinematic constraints:* To guarantee that the vehicles speeds and accelerations are within admissible range the following constraints are defined:

$$a_j(t) \leq a_{\max}, \quad 0 \leq v_j(t) \leq v_{\max}, \quad j = 1, \dots, k, \quad (6)$$

with a_j being the total acceleration of vehicle j , v_j the speed of vehicle j , and a_{\max} and v_{\max} the maximum total acceleration and maximum speed, respectively.

2) *Vehicle-to-vehicle collision avoidance constraints:* The distance, $d_{ij}(t)$, between every two vehicles i and j must be kept at or above a minimum safe distance, d_s :

$$d_{ij}(t) \geq d_s, \quad i = 1, \dots, k, \quad j = 1, \dots, k, \quad i < j. \quad (7)$$

3) *Plaza boundaries constraints:* Violation of the plaza boundaries by the CVAD must be disallowed. Hence, the intersection's geometric constraints, based on (1), that ensure there are no collisions of CVAD with the boundaries, are:

$$\begin{cases} y_j(t) \leq f_h(x_j(t)), & \text{if } h = 1, 2 \\ y_j(t) \geq f_h(x_j(t)), & \text{if } h = 3, 4 \end{cases} \quad j = 1, \dots, k, \quad (8)$$

with $(x_j(t), y_j(t))$ being the position of vehicle j in the CCS.

D. Performance Index

We choose as the performance index a weighted sum of Δv and T that should be minimized:

$$\mathcal{J} = w_1 \cdot \Delta v + w_2 \cdot T, \quad (9)$$

with $w_1 \geq 0$ and $w_2 \geq 0$ weighting parameters. Minimizing Δv and T are conflicting objectives that affect fuel consumption and comfort (since speed increment is minimized) versus speed and maneuver completion time.

E. Intersection trajectories optimal control problem

By considering the EoM, the constraints, and the performance index, the ITOP can be formulated as:

$$\text{minimize } \mathcal{J} = w_1 \cdot \Delta v + w_2 \cdot T, \quad (10)$$

subject to (6)–(8) with $h = 1, \dots, 4$, and $0 \leq t \leq T$.

By appropriately writing the EoM in state-space representation, the ITOP formulation can be expressed as a non-linear optimal control problem [11]. Since the detailed formulation is not needed for the approach in the next section, it is omitted here. We note, however, that the states are the position and speed of each vehicle j in the CCS ($x_j(t)$, $y_j(t)$, $v_{x_j}(t)$, and $v_{y_j}(t)$), and the decision variables are $a_{x_j}(t)$, $a_{y_j}(t)$, and T .

III. OPTIMIZATION PROBLEM

In this section, we propose the Bézier curve method [22] for solving the ITOP for the CVAD at the plaza. In this method, the state variables, i.e., positions and speeds, are interpolated, and control variables, i.e., accelerations and completion time, are considered in the objective function. Then, the Bézier representation of state variables is imposed on the dynamics, and the required accelerations to realize the resulting trajectories are evaluated. Finally, the ITOP is reduced to a system of algebraic equations in the Bézier coefficients and a collision-free trajectory optimization problem is formulated.

A. Bézier Approximations

The Bézier curves have several properties for trajectory optimization that are appropriate for the purpose of this work: (i) the starting and ending points of the curve correspond to the first and final Bézier coefficients, respectively; (ii) the curve completely lies within the convex hull formed by all Bézier coefficients; and (iii) the curves have the advantage of simplicity and curvature continuity.

In this part, a Bézier curve is employed to approximate each position state variable of each vehicle j in each coordinate of the CCS (x_j and y_j) as follows [22]:

$$\mathbf{z}(\tau) = \sum_{l=0}^{n_z} B_{\mathbf{z},l}(\tau) P_{\mathbf{z},l}, \quad (11)$$

with $\mathbf{z} = [x_j(\tau) \ y_j(\tau)]^T$, $0 \leq \tau = t/T \leq 1$ is the scaled time, n_z is the number of Bézier terms (order of the Bézier curve), $P_{\mathbf{z},l}$ are the unknown Bézier coefficients to be determined, and $B_{\mathbf{z},l}(\tau)$ are the Bernstein basis polynomials given by:

$$B_{\mathbf{z},l}(\tau) = \binom{n_z}{l} \tau^l (1-\tau)^{n_z-l}, \quad l \in \{0, 1, \dots, n_z\}. \quad (12)$$

The speed state variables of each vehicle j in each coordinate of the CCS (v_{x_j} and v_{y_j}) are the first derivatives of (11) with respect to the scaled time (τ):

$$\mathbf{z}'(\tau) = \sum_{l=0}^{n_z} B'_{\mathbf{z},l}(\tau) P_{\mathbf{z},l}, \quad (13)$$

with

$$B'_{\mathbf{z},l}(\tau) = \begin{cases} -n_z(1-\tau)^{n_z-1}, & \text{if } l = 0, \\ \frac{n_z! \tau^{l-1} (1-\tau)^{n_z-l}}{(l-1)! (n_z-l)!} - \frac{n_z! \tau^l (1-\tau)^{n_z-l-1}}{l! (n_z-l-1)!}, & \text{if } l \in [1, n_z-1], \\ n_z \tau^{n_z-1}, & \text{if } l = n_z. \end{cases} \quad (14)$$

1) *Boundary conditions:* For each vehicle, we know the Boundary Conditions (BCs), i.e., the initial and final positions and speeds in the coordinate system. The BCs with respect to the scaled time for each vehicle are:

$$\mathbf{z}(0) = \mathbf{z}_i, \quad \mathbf{z}(1) = \mathbf{z}_f, \quad \mathbf{z}'(0) = T \dot{\mathbf{z}}_i, \quad \mathbf{z}'(1) = T \dot{\mathbf{z}}_f. \quad (15)$$

The labels ‘i’ and ‘f’ refer to ‘initial’ and ‘final’, respectively, the prime denotes the derivative with respect to the scaled time and the dot the derivative with respect to the continuous time. These relations are obtained through the chain rule [16].

2) *Using the BCs for expressing some coefficients:* By manipulating algebraically (11)–(15), it is straightforward to derive four Bézier coefficients of (11) as:

$$\begin{aligned} P_{\mathbf{z},0} &= \mathbf{z}_i, & P_{\mathbf{z},1} &= \mathbf{z}_i + \frac{T \dot{\mathbf{z}}_i}{n_z}, \\ P_{\mathbf{z},n_z-1} &= \mathbf{z}_f - \frac{T \dot{\mathbf{z}}_f}{n_z}, & P_{\mathbf{z},n_z} &= \mathbf{z}_f, \end{aligned} \quad (16)$$

reducing the number of unknown Bézier coefficients, thus speeding up the optimization. Substituting these coefficients in (11) and organizing the resulting expression gives:

$$\mathbf{z}(\tau) = F_{\mathbf{z}} + \sum_{l=2}^{n_z-2} B_{\mathbf{z},l}(\tau) P_{\mathbf{z},l}, \quad (17)$$

with

$$F_{\mathbf{z}} = B_{\mathbf{z},0} P_{\mathbf{z},0} + B_{\mathbf{z},1} P_{\mathbf{z},1} + B_{\mathbf{z},n_z-1} P_{\mathbf{z},n_z-1} + B_{\mathbf{z},n_z} P_{\mathbf{z},n_z}. \quad (18)$$

The corresponding first and second derivatives with respect to the scaled time, $\mathbf{z}'(\tau)$ and $\mathbf{z}''(\tau)$, can be readily obtained.

3) *Evaluation points*: In order to solve for the unknown Bézier coefficients, the EoM are evaluated at m points, called Discretization Points (DPs). We consider m DPs with equal time intervals within the scaled time ($\tau_i - \tau_{i-1} = 1/(m-1), i = 2, \dots, m$):

$$\tau_1 = 0 < \tau_2 < \dots < \tau_{m-1} < \tau_m = 1. \quad (19)$$

The constraints in Section II-C are satisfied only at each DP. Thus, to avoid violations in-between DPs, we must choose a large enough safe distance and/or sufficiently dense DPs.

4) *Compact matrix form representation*: Since the EoM are evaluated at the DPs, a compact matrix form representation for the position state variables and its derivatives (speed state variables and accelerations) already incorporating the coefficients from the BCs can be used. We can write the position state variables at the m DPs as vectors of its values:

$$[\mathbf{z}]_{m \times 1} = [B_{\mathbf{z}}]_{m \times (n_{\mathbf{z}}-3)} [X_{\mathbf{z}}]_{(n_{\mathbf{z}}-3) \times 1} + [F_{\mathbf{z}}]_{m \times 1}, \quad (20)$$

with $[F_{\mathbf{z}}]$ being a constant vector depending on $n_{\mathbf{z}}$ and on the BCs obtained using (18), $[B_{\mathbf{z}}]$ being a matrix of coefficients given by:

$$[B_{\mathbf{z}}]_{m \times (n_{\mathbf{z}}-3)} = [B_{\mathbf{z},2} \ B_{\mathbf{z},3} \ \dots \ B_{\mathbf{z},n_{\mathbf{z}}-2}]^T, \quad (21)$$

and $[X_{\mathbf{z}}]$ being the vector of unknown Bézier coefficients:

$$[X_{\mathbf{z}}]_{(n_{\mathbf{z}}-3) \times 1} = [P_{\mathbf{z},2} \ P_{\mathbf{z},3} \ \dots \ P_{\mathbf{z},n_{\mathbf{z}}-2}]^T. \quad (22)$$

Matrices $[B_{\mathbf{z}}]$ and $[F_{\mathbf{z}}]$ are computed offline and $[X_{\mathbf{z}}]$ results from the optimization. The compact forms of the first and second derivatives of (20) have a similar structure.

Then, the total acceleration of vehicle j along its trajectory can be represented in a compact matrix form as well by replacing (20) and its derivatives in (2) and (3):

$$[a_j]_{m \times 1} = \sqrt{\sum_{\forall \mathbf{z}} [a_{\mathbf{z}}]_{m \times 1}^2} \leq [a_{\max}]_{m \times 1}. \quad (23)$$

Note that $a_{\mathbf{z}} = \ddot{\mathbf{z}}$, therefore, we need the relations between $\ddot{\mathbf{z}}$ and \mathbf{z}'' via the chain rule (see Section III-A1).

B. Nonlinear Programming Formulation

Given the compact matrix form (20) and corresponding derivatives, we can formulate a NLP with the unknown Bézier coefficients $[X_{\mathbf{z}}]$ and the completion time T as decision variables:

$$\begin{aligned} & \min_{[X_{\mathbf{z}}] \forall \mathbf{z} \forall j, T} \mathcal{J} \\ & \text{s.t. } [a_j(t)] \leq [a_{\max}], \\ & \quad 0 \leq [v_j(t)] \leq [v_{\max}], \\ & \quad [d_{ij}(t)] \geq [d_s], \\ & \quad [y_j(t)] \leq [f_h(x_j(t))], \text{ if } h = 1, 2 \\ & \quad [y_j(t)] \geq [f_h(x_j(t))], \text{ if } h = 3, 4, \end{aligned} \quad (24)$$

with $i = 1, \dots, k$, $j = 1, \dots, k$, and $i < j$. We note that (20) and its derivatives are embedded in formulation (24) through the substitution in (2)–(4).

TABLE I: SETTINGS FOR VEHICLES/ INTERSECTIONS

| Parameter | Description | Value |
|-------------------|--|-------|
| k | Number of CVAD | 3 |
| a_{\max} | Maximum acceleration (m/s ²) | 2 |
| v_{\max} | Maximum speed (m/s) | 10 |
| d_s | Safe distance (m) | 1 |
| W_{road} | Road width (m) | 11 |
| L_{road} | Road length (m) | 90 |

C. Initialization of Decision Variables

The initialization of the unknown Bézier coefficients can be expressed in a compact form as:

$$[X_{\mathbf{z}}]_{(n_{\mathbf{z}}-3) \times 1} = \left([B_{\mathbf{z}}]_{n_a \times (n_{\mathbf{z}}-3)} \right)^{-1} ([\mathbf{z}_a]_{n_a \times 1} - [F_{\mathbf{z}}]_{n_a \times 1}), \quad (25)$$

with n_a being the number of DPs for the approximation and $[\mathbf{z}_a]$ the approximated position state variables. A cubic Bézier curve can be used to approximate $[\mathbf{z}_a]$ using the BCs [16][19].

The initialization of the completion time T_a can be approximated by arbitrarily selecting the time taken by a vehicle to cross in a straight direction from its origin to its destination with maximum total acceleration as:

$$T_a = \sqrt{\frac{2S}{a_{\max}}}, \quad (26)$$

where S is the distance between the origin and destination of the selected vehicle.

IV. NUMERICAL RESULTS

In this section, we evaluate the solution of the ITOP via the Bézier curve method with MATLAB 2018b. The NLP is solved using the fmincon solver of the optimization toolbox. Each $\Delta v_j(t)$ in (4) is computed by numerical integration of the corresponding total acceleration over time T through the built-in function trapz. Moreover, we compare the results obtained with the Bézier method with the approach based on the FFS [15].

A. Scenario setup

We investigate a scenario with three CVAD at the plaza. The goal is to show that the proposed Bézier method is able to generate near-optimal collision-free trajectories for these three vehicles. CVAD₁ travels from north to east, CVAD₂ travels from south to west, and CVAD₃ goes straight from east to west. The center of the plaza is the origin of the CCS. The initial positions (x_{ij}, y_{ij}) of the three CVAD are $(-2, 40)$ m, $(2, -40)$ m, and $(43, 8)$ m, respectively, and the final positions (x_{fj}, y_{fj}) are, respectively, $(45, -4)$ m, $(-45, 1)$ m, and $(-45, 8)$ m. The initial speeds (v_{ix_j}, v_{iy_j}) are $(1, -5)$ m/s, $(-1, 5)$ m/s, and $(-7, 0)$ m/s, respectively, and the final speeds (v_{fx_j}, v_{fy_j}) are, respectively, $(6, -2)$ m/s, $(-6, 2)$ m/s, and $(-8, 0)$ m/s. The required parameters for the formulation are reported in Tables I and II.

The results discussed next and summarized in Table III were obtained by the solution of the NLP (24) with $w_1 = 5$, $w_2 = 2$, $m = 30$ and $n_{\mathbf{z}} = 8$ for the Bézier method and with $w_1 = 4$, $w_2 = 2$, $m = 30$ and $n_{\mathbf{z}} = 6$ for the FFS method in [15].

TABLE II: SCALING PARAMETERS

| Parameter | IB ₁ | IB ₂ | IB ₃ | IB ₄ |
|-----------|-----------------|-----------------|-----------------|-----------------|
| r_0 | 11 | 11 | -11 | -11 |
| r_1 | 1 | 1 | -1 | -1 |
| r_2 | -1 | 1 | 1 | -1 |
| r_3 | -11 | 11 | 11 | -11 |

TABLE III: NUMERICAL RESULTS OF BÉZIER AND FFS

| Method | Δv (m/s) | T (s) | J | T_c (s) |
|--------|------------------|---------|------|-----------|
| Bézier | 14.1 | 11.5 | 93.5 | 4.3 |
| FFS | 18.2 | 12.3 | 97.4 | 7.9 |

B. Quantitative Results

As shown in Table III, the computation time (T_c) of the Bézier method is lower than the one of the FFS method due to the smaller number of decision variables in the first method. Moreover, smaller total speed increment and completion time were obtained with the Bézier method.

We experimented with different combinations of values for m , w_1 , w_2 and n_z that were considered by trial and error (details not shown). Varying the values of w_1 and w_2 had more influence on the total speed increment than the completion time. When the number of DPs (m) is increased, there is an increase in computation time. Despite the corresponding increase in total speed increment, completion times also increase, suggesting that worse local minima are found for higher values of m , i.e., trajectories in longer paths result. Finally, increasing n_z increases the computation time without sensible improvements in the other measures. Small values of m and n_z may result in better values of Δv , T , and computation time. However, the trajectories might not be smooth and may also lead to infeasible instances of the NLP problem, as also observed with the FFS method [15].

C. Analysis of the Trajectories

The optimized trajectories generated by the solution of the NLP based on the Bézier and FFS methods are illustrated in Fig. 3. The colored disks corresponding to the cool colormap indicate DPs and time of the Bézier method (T_B), and the colored disks corresponding to the warm colormap indicate DPs and time of the FFS method (T_F). The gray squares are the initial positions for each CVAD. The solid thick black lines show the boundary of each IB whose approximations are presented by black dashed lines. For this particular scenario, the trajectories of both methods deviate from what would be expected in a movement-based method and it is clear that the CVAD follow free trajectories. Noteworthy, the followed paths of the Bézier method seem to approach the paths of minimum distance compared to the FFS method.

Fig. 4 shows the distances between every two CVAD generated with the Bézier method. It can be observed that, at all times, a minimum safe distance (d_s) is maintained between all CVAD. Accordingly, the vehicles distances remain above d_s by a large margin for this scenario. Similar results were observed for the FFS case (not shown) [15].

Figs. 5(a), (c), and (e) show the total (absolute) acceleration (a_j), and the acceleration (\hat{a}_j) for each of the three CVAD,

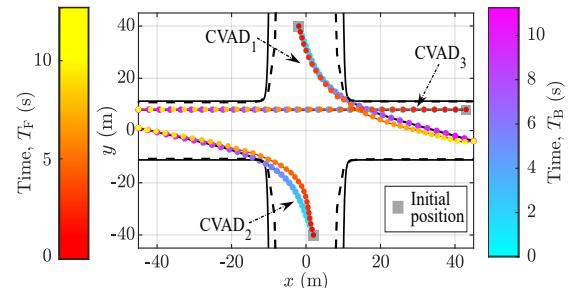


Fig. 3: Trajectories of the CVAD using Bézier method (T_B) and FFS method (T_F).

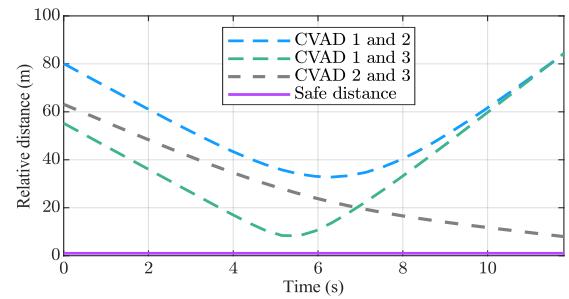


Fig. 4: Distance between every two CVAD using the Bézier method and $d_s = 1$ m.

with $j = 1, 2, 3$, for both Bézier and FFS methods. Figs. 5(b), (d), and (f) show the speed (v_j) for the same three vehicles. The total acceleration and speed profiles of each vehicle are far below the maximum total acceleration and maximum speed values, respectively, satisfying the constraints. In addition, the profiles of both methods are smooth, as expected due to the minimization of Δv . However, the acceleration and speed profiles of the trajectories generated with the FFS method exhibit more variation compared to the ones with the Bézier method. Accordingly, we can conclude that the Bézier method is capable of providing more comfortable vehicle movement with less computation time.

To evaluate the efficacy of the collision avoidance constraint, we present in Fig. 6 the numerical results of three CVAD based on a larger value of safe distance, $d_s = 7$ m. We observe in Fig. 6a a slightly different behavior of CVAD₁, which keeps more distance from the corner of the IB (compare to Fig. 3) due to the activation of collision avoidance constraints. It can be seen in Fig. 6b that the collision avoidance constraint avoids the collision between CVAD₁ and CVAD₃ at around $t = 5$ s, and subsequently, the distance between the vehicles (green dashed line) does not go below the safe distance line (solid purple line).

V. CONCLUSION

This paper introduced the ITOP with free use of the intersection space, called a plaza, by CVAD. There are no vehicle movement-related constraints except for the intersection boundaries, which were properly modeled. The Bézier curve method and discretization were used to transform the

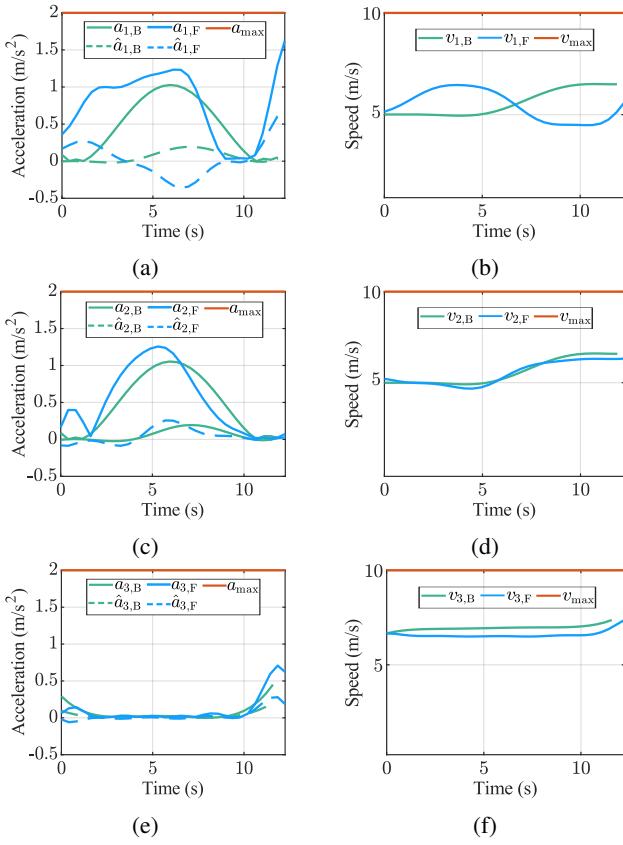


Fig. 5: (a), (c), and (e) Acceleration profiles and (b), (d), (f) speed profiles of j -th CVAD, with $j = 1, 2, 3$.

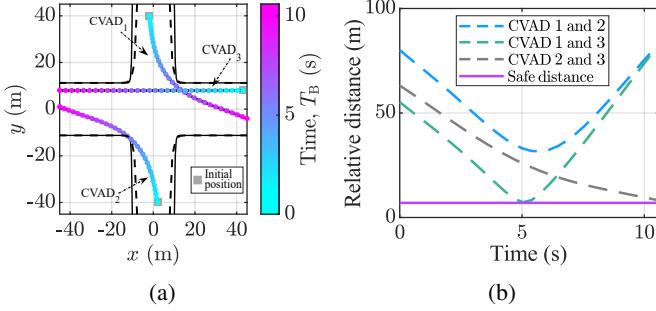


Fig. 6: (a) Trajectories of the CVAD using Bézier method; and (b) distance between them with $d_s = 7$ m.

ITOP into a nonlinear program problem. The method is able to generate near-optimal collision-free trajectories for the CVAD coordination at the plaza. Finally, the results of the proposed method were compared to the FFS method, showing slightly better results.

Future work should consider more elaborate vehicle dynamics, vehicle dimensions, different completion times for the CVAD, and free final states. Other objectives should be evaluated and additional constraints should be added, e.g., bounds on jerk values. The continuous arrival of vehicles should be handled along with a comparison with movement-based approaches based on typical traffic performance metrics,

such as traffic delay and intersection capacity.

ACKNOWLEDGMENT

We acknowledge partial financial support from the CAPES Foundation, Ministry of Education of Brazil, and the National Council for Scientific and Technological Development (CNPq).

REFERENCES

- [1] L. Chen and C. Englund, "Cooperative Intersection Management: A Survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 2, pp. 570–586, 2016.
- [2] J. Rios-Torres and A. A. Malikopoulos, "A Survey on the Coordination of Connected and Automated Vehicles at Intersections and Merging at Highway On-ramps," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 5, pp. 1066–1077, 2017.
- [3] E. R. Müller, "Optimal Arrival Time Scheduling of Automated Vehicles at Intersections," Doctoral dissertation, Post-graduate Program in Automation and Systems Engineering, Federal University of Santa Catarina, Florianópolis, SC, Brazil, 2018.
- [4] K. Dresner and P. Stone, "A Multiagent Approach to Autonomous Intersection Management," *Journal of Artificial Intelligence Research*, vol. 31, pp. 591–656, 2008.
- [5] H. Schepperle and K. Böhm, "Valuation-aware Traffic Control: The Notion and the Issues," in *Multi-agent Systems for Traffic and Transportation Engineering*. IGI Global, 2009, pp. 218–239.
- [6] J. Lee and B. Park, "Development and Evaluation of a Cooperative Vehicle Intersection Control Algorithm under the Connected Vehicles Environment," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 1, pp. 81–90, 2012.
- [7] F. Zhu and S. V. Ukkusuri, "A Linear Programming Formulation for Autonomous Intersection Control within a Dynamic Traffic Assignment and Connected Vehicle Environment," *Transportation Research Part C: Emerging Technologies*, vol. 55, pp. 363–378, 2015.
- [8] E. R. Müller, R. C. Carlson, and W. Kraus, "Intersection Control for Automated Vehicles with MILP," *IFAC-PapersOnLine*, vol. 49, no. 3, pp. 37–42, 2016.
- [9] M. W. Levin and D. Rey, "Conflict-point Formulation of Intersection Control for Autonomous Vehicles," *Transportation Research Part C: Emerging Technologies*, vol. 85, pp. 528–547, 2017.
- [10] B. Li, Y. Zhang, Y. Zhang, N. Jia, and Y. Ge, "Near-optimal Online Motion Planning of Connected and Automated Vehicles at a Signal-free and Lane-free Intersection," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, 2018, pp. 1432–1437.
- [11] 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*, vol. 53, no. 2, pp. 17071–17076, 2020.
- [12] D. Roca, D. Nemirovsky, M. Nemirovsky, R. Milito, and M. Valero, "Emergent Behaviors in the Internet of Things: The Ultimate Ultra-large-scale System," *IEEE Micro*, vol. 36, no. 6, pp. 36–44, 2016.
- [13] D. Roca, R. Milito, M. Nemirovsky, and M. Valero, "Advances in the Hierarchical Emergent Behaviors (HEB) Approach to Autonomous Vehicles," *IEEE Intelligent Transportation Systems Magazine*, vol. 12, no. 4, pp. 57–65, 2020.
- [14] G. J. Ninos Neto, "Analysis of the Management of Automated Vehicles with Free Exploration of the Intersection Area (in Portuguese)," Master thesis, Post-graduate Program in Automation and Systems Engineering, Federal University of Santa Catarina, Florianópolis, SC, Brazil, 2021.
- [15] E. Ahmadi, R. C. Carlson, W. Kraus Junior, and E. Taheri, "Near-optimal Coordination of Vehicles at an Intersection Plaza using Finite Fourier Series," in *35º Congresso Nacional de Pesquisa e Ensino em Transportes*. Brazil: ANPET, 2021.
- [16] E. Taheri and O. Abdelkhalik, "Initial Three-dimensional Low-thrust Trajectory Design," *Advances in Space Research*, vol. 57, no. 3, pp. 889–903, 2016.
- [17] R. Lattarulo and et al., "Urban Motion Planning Framework based on n-Bézier Curves Considering Comfort and Safety," *Journal of Advanced Transportation*, vol. 2018, 2018.
- [18] M. Schwung and J. Lunze, "Networked Event-Based Collision Avoidance of Mobile Objects with Trajectory Planning based on Bézier Curves," *European Journal of Control*, vol. 58, pp. 327–339, 2021.

- [19] Z. Fan, M. Huo, S. Xu, J. Zhao, and N. Qi, "Fast Cooperative Trajectory Optimization for Close-Range Satellite Formation Using Bezier Shape-Based Method," *IEEE Access*, vol. 8, pp. 30918–30927, 2020.
- [20] B. Siciliano and O. Khatib, Eds., *Springer Handbook of Robotics*. Berlin, Heidelberg: Springer, 2008.
- [21] A. G. Ulsoy, H. Peng, and M. Cakmakci, *Automotive Control Systems*. Cambridge: Cambridge University Press, 2012.
- [22] R. T. Farouki, *Pythagorean—hodograph Curves*. Springer, 2008.