This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2021.3093280, IEEE Transactions on Automatic Control

# Guaranteed Collision-Free Reference Tracking in Constrained Multi Unmanned Vehicle Systems

Maryam Bagherzadeh, Shima Savehshemshaki, Walter Lucia

***Abstract*—In this paper, we face the reference tracking control problem for a system of heterogeneous Multi Unmanned Vehicles (MUVs) moving in a 2-D planar environment. We consider a scenario where each vehicle follows a trajectory imposed by a local planner and where each UV can have different linear dynamics as well as different constraints and disturbances. In this contest, we design a novel control architecture where a centralized traffic manager, in conjunction with ad-hoc designed local vehicle controllers, is capable of ensuring the absence of collisions. The proposed solution is obtained by exploiting, for the local vehicles' controllers, a dual-mode Model Predictive Controller (MPC) and, for the traffic manager, set-theoretic and controllability properties. Moreover, after modeling the potential vehicle collisions with a graph, connectivity arguments are used to obtain an optimal collision resolution which minimizes the number of vehicles that need to be stopped. The resulting control scheme ensures collision-free signal tracking. Simulation results, conducted on a MUV system are shown to provide tangible evidence of the features of the proposed framework.**

***Index Terms*—Collision avoidance; Traffic control; Constrained control; Robust control; Autonomous vehicles.**

## I. Introduction

To obtain a fully automated transportation system, different issues must be addressed [1]. Each vehicle must be able to track a reference signal and reach the desired goal [2], [3], formation control tools are needed to guarantee coordination among vehicles [4], [5], and absence of collisions must be ensured [6]. Of interest for this paper are control solutions capable of addressing the collision-free reference tracking control problem for MUVs moving in a 2D planar environment. In [5], [7]–[12], different multi-agent strategies have been proposed to address the collision avoidance problem. Such approaches, although appealing, cannot be straightforwardly used to take into account the simultaneous presence of state and input constraints and exogenous disturbances. In [13], collision-free movements and deadlock-avoidance for multi-robot systems are achieved by defining a proper quadratic programming problem based on a mixture of relaxed control barrier functions, hybrid braking controllers, and consistent perturbations. In [14], the collision avoidance problem for a group of vehicles is formulated as a Mixed Integer Linear Programming (MILP) problem where each vehicle has an apriori known fixed number of possible trajectories. Along similar lines is also the work in [15] where the exponential complexity of the obtained Hamilton-Jacobi (HJ) and MILP problems is reduced using a combinational technique based on HJ reachability and MILP programming concepts. In [16], a robust Decentralized Model Predictive Controller (DMPC) for a team of UVs is proposed. In particular, absence of collisions are modeled as coupling constraints and the resulting centralized MILP optimization problem is recast into smaller size MILP problems, sequentially solved by the vehicles. Such a solution, at the cost of increasing communication requirements among the vehicles and conservativeness, is proved to better scale with the number of vehicles with respect to the centralized counterpart. In [17], it has been shown that in structured environments, with a finite and apriori known number of trajectories, the inter-vehicles communication demand of decentralized solutions can be reduced by resorting to a centralized entity in charge of coordinating the vehicles trajectories.

Starting from the discussed state-of-the art, in this paper, we introduce a novel control solution, based on set-theoretic arguments [18]–[20], capable of dealing with the MUV collision-free reference tracking problems. We assume a general setting where each vehicle is equipped with a reference generator module and a local controller. Moreover, the vehicles' trajectories are neither coordinated nor shared with the other vehicles. The solution here presented is based on two main ingredients: local set-theoretic reference tracking controllers and a centralized collision-avoidance supervisor (Fig. 1). It is shown that the local controllers require the computation of quadratic programming problems and the collision avoidance logic is based on simple set-membership tests naturally arising from the employed local controllers. Contrary to the solution in [17], the environment does not need to be structured, and the collision-avoidance supervisor does not need to know all the possible vehicle trajectories. Differently from existing DMPC approaches, the proposed solution solves in a decentralized fashion the reference tracking problem while leaving centralized only the collision avoidance. As a consequence, the local controllers solve independent convex programming optimization problems (instead of sequential MILP, see e.g., [16]), and each vehicle does not need to exchange its trajectory with the other vehicles. Although we cannot claim any improvement in terms of solution conservativeness with respect to the state-of-the-art, the proposed control architecture presents some peculiar capabilities. First, it solves the reference tracking and the collision avoidance problems without requiring the solution of non-convex optimization problems. Second, it does not require inter-vehicle communications, and as a consequence, it is well-suited for smart city contests where absence of vehicle communications is desirable for privacy reasons [21].

The present work extends and improves the preliminary solution in [22] in the following main aspects: the control scheme has been extended to face MUVs with a finite arbitrary number of vehicles; a new collision avoidance algorithm has been proposed to guarantee optimal collision avoidance by resorting to graph connectivity theory; all the main results and strategy properties have been formally proved (no proofs are provided in [22]); and a more exhaustive simulation example has been provided.

## II. Problem Formulation

Let us consider UVs moving in a two-dimensional planar environment of coordinates $p_i = [p_i^x, p_i^y]^T$.

*Assumption 1:* The vehicle's dynamics can be described by the class of constrained discrete-time linear time-invariant (LTI) systems subject to additive bounded exogenous disturbances.

The state space representation for the $i$-th vehicle dynamic, namely $UV_i$, is:

$$x_i(t+1) = A_i x_i(t) + B_i u_i(t) + d_i(t), \;\; p_i(t) = C_i x_i(t) \quad (1)$$

This work was supported in part by the Natural Sciences and Engineering Council of Canada under Grant RGPIN-2018-06713 and in part by the Fonds Québécois de la Recherche sur la Nature et les Technologies under Grant 2020-NC-268119.

M. Bagherzadeh, S. Savehshemshaki and W. Lucia are with the CIISE Department, Concordia University, Montreal, H3G-1M8, Canada (e-mails: m_baherz@encs.concordia.ca, s_savehs@encs.concordia.ca, walter.lucia@concordia.ca).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2021.3093280, IEEE Transactions on Automatic Control

where $t \in \mathbb{Z}_+ := \{0, 1, ...\}$ is the sampling time instants, $u_i \in \mathbb{R}^{m_i}$ the control input, $x_i = [p_i^T, z_i^T]^T \in \mathbb{R}^{n_i}$ the state space vector with $z_i \in \mathbb{R}^{n_i-2}$ the vector of non-spatial state-space variables (e.g., vehicle velocities). $A_i$, $B_i$ are matrices of suitable dimensions, $C_i$ is a matrix which selects the spatial components $p_i$ from the state-vector $x_i$ and $d_i$ is a bounded exogenous disturbance such that

$$d_i(t) \in \mathcal{D}_i \subset \mathbb{R}^{d_i}, \quad 0_{d_i} \in \mathcal{D}_i \tag{2}$$

with $\mathcal{D}_i$ a compact set. The following state and input constraints are prescribed

$$u_i(t) \in \mathcal{U}_i, \quad x_i(t) \in \mathcal{X}_i = \mathbb{R}^2 \times \mathcal{Z}_i, \ \forall t \geq 0, \tag{3}$$

where $\mathcal{U}_i \subseteq \mathbb{R}^{m_i}$ and $\mathcal{Z}_i \subseteq \mathbb{R}^{n_i-2}$ are compact subsets with $0_{m_i} \in \mathcal{U}_i$ and $0_{n_i-2} \in \mathcal{Z}_i$, respectively.

*Assumption 2:* The vehicle's model (1) is reachable and it includes two integral effects, i.e., $A = \begin{bmatrix} \mathbf{I} & A_{12} \\ \mathbf{0} & A_{22} \end{bmatrix}$, with $\mathbf{I} \in \mathbb{R}^{2\times 2}$ and $\mathbf{0} \in \mathbb{R}^{n_i-2\times 2}$ the identity and zero matrix, respectively, and $A_{12} \in \mathbb{R}^{2\times n_i-2}$, $A_{22} \in \mathbb{R}^{n_i-2\times n_i-2}$. Therefore, the pair $(x_{p_i}^{eq} := [p_i^T, 0_{n_i-2}]^T, u_i^{eq} = 0_{m_i})$ defines, for any $p_i \in \mathbb{R}^2$, an equilibrium point for $UV_i$.

*Definition 1:* Let $\mathcal{Q} \subset \mathbb{R}^n$ be a neighborhood region of the origin. The closed-loop trajectory of (1), is Uniformly Ultimately Bounded (UUB) in $\mathcal{Q}$ if for all $\mu > 0$, there exists a function $T(\mu) > 0$ such that $\forall \|x(0)\| \leq \mu \rightarrow x(t) \in \mathcal{Q}, \forall d_i(t) \in \mathcal{D}_i$ and $\forall\, t \geq T(\mu)$ [19].

*Definition 2:* A set $\mathcal{Q} \subseteq \mathcal{X}_i$ is said to be Robust Control Invariant (RCI) for (1) under (2)-(3) if $\forall x_i(t) \in \mathcal{Q} \rightarrow \exists u_i(t) \in \mathcal{U}_i : A_i x_i(t) + B_i u_i(t) + d_i(t) \in \mathcal{Q}, \ \ \forall d_i(t) \in \mathcal{D}_i, \forall t \in \mathbb{Z}_+$ [23].

### A. Considered Setup

*UV models*: each $i$-th UV is described by (1)-(3). Each UV can have a different model in terms of $(A_i,\ B_i,\ C_i)$ matrices and/or constraints $(\mathcal{X}_i, \mathcal{U}_i)$ and/or disturbance sets $(\mathcal{D}_i)$. Moreover, we claim absence of collisions between $UV_i$ and $UV_j$ if $||p_i(t) - p_j(t)||_2 > 0, \ \ \forall t \in \mathbb{Z}_+$.

*Reference generators*: each UV is equipped with a reference generator performing trajectory planning at a kinematic level without taking into account the complete vehicles' model, constraints, disturbances, and the presence of other vehicles. The reference generators provide the trajectories $\mathbf{r}_i$ to the vehicles as a sequence of waypoints $r_i(k_i) \in \mathbb{R}^2$ indexed by $k_i \in \mathbb{Z}_+$. We assume there exists a maximum distance $\delta_i > 0$ between two successive points, i.e. $||r_i(k_i+1) - r_i(k_i)||_2 \leq \delta_i$.

*Communication facilities*: Inter-vehicle communications are not possible, but each vehicle is able to share information with a centralized unit hereafter referred to as the traffic manager.

***Remark 1:*** In the sequel, the discrete index $t$ denotes the discrete-time evolution of the vehicles' variables, e.g. $x_i(t)$, $u_i(t)$, and $p_i(t)$, the index $k_i$ denotes the $k_i - th$ waypoint $r_i(k_i)$ generated by the reference generator.

The control problem of interest can be stated as follows:

***UVs Reference Tracking with Guaranteed Collision Avoidance***: Given a set $\mathcal{I} := \{1, 2, \ldots, S\}$ of $S$ heterogeneous unnamed vehicles which models are described by (1)-(3):

- (**O1**) Design decentralized state-feedback controllers $u_i(t) = f_i(x_i(t), r_i(k_i))$ fulfilling constraints (3) regardless disturbance realizations (2) and able to sequentially track the waypoints $r_i(k_i), \forall k_i$, with a tracking error that, for each waypoint, is UUB in a finite number of steps.
- (**O2**) Design a centralized traffic manger module capable of guaranteeing absence of collisions among vehicles regardless of the reference trajectories $\mathbf{r}_i$, $i \in \mathcal{I}$.

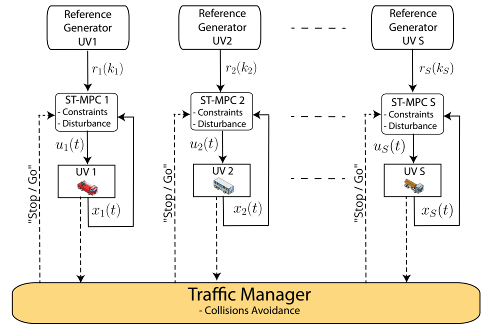


Fig. 1. Proposed Control Architecture

## III. Proposed Solution

The proposed control architecture, illustrated in Fig. 1, consists of a set of $S$ set-theoretic model predictive controllers (**ST-MPC**) which takes care of the tracking problem (**O1**), and a centralized Traffic Manager (**TM**) that solves the collision avoidance problem (**O2**).

### A. Set-Theoretic MPC Tracking Controller

In this subsection, we provide a solution to the control objective (**O1**). Our approach is based on the dual-mode set-theoretic MPC (ST-MPC) paradigm inspired by the works done in [19], [20], [24]. In particular, the proposed tracking control strategy consists of two main actions: (*i*) regulation towards the current waypoint $r_i(k_i)$ and (*ii*) waypoint update $r_i(k_i) \rightarrow r_i(k_i+1)$.

***(i) Regulation towards the current waypoint***: Let's first consider the $i$-th UV and a waypoint $r_i(k_i)$, and denote with $(x_{r_i(k_i)}^{eq} = [r_i^T(k_i), 0_{n_i-2}]^T, u_{r_i(k_i)}^{eq} = 0)$ the equilibrium pair corresponding to the waypoint $r_i(k_i)$. The constrained waypoint regulation problem is solved according to the prescription of ST-MPC scheme in [20]. The offline (*Off-1* and *Off-2*) and online (*On-1* and *On-2*) steps of the strategy are here summarized:

*–Offline–*

*-(Off-1)* Given (1), design a stabilizing state-feedback control law $u_i(t) := f_i^0(x_i(t), x_{r_i(k_i)}^{eq})$ and the associated smallest RCI region, namely $\mathcal{T}_i^0(r_i(k_i))$ [25]. The region $\mathcal{T}_i^0$ is hereafter referred to either as the terminal region or as the domain of attraction of the terminal controller, namely $DoA_i^0$.

*-(Off-2)* The domain of the controller found in the previous step is enlarged to ensure that any initial state $x_i(t)$ belongs to the controller admissible region. The latter is obtained by computing a family of $N_i$ robust one-step controllable sets, namely $\{\mathcal{T}_i^l\}_{l=0}^{N_i}$, $N_i \geq 1$, by applying the following recursive definition for robust one-step controllable sets [19]:

$$\begin{aligned} \mathcal{T}_i^l \ &:= \ \{x_i \in \mathcal{X}_i : \exists u_i \in \mathcal{U}_i : \forall d_i \in \mathcal{D}_i, A_i x_i + B_i u_i + d_i \in \mathcal{T}_i^{l-1}\} \\ &= \ \{x_i \in \mathcal{X}_i : \exists u_i \in \mathcal{U}_i : A_i x_i + B_i u_i \in \tilde{\mathcal{T}}_i^{l-1}\} \end{aligned} \tag{4}$$

where $N_i$ is the number of computed sets, and $\tilde{\mathcal{T}}_i^{l-1} := \mathcal{T}_i^{l-1} \sim \mathcal{D}_i$ with $\sim$ denoting the Minkowski set-difference operator [23]. The set union $\bigcup_{l=0}^{N_i}\{\mathcal{T}_i^l\}$ defines the controller domain of attraction, namely $DoA_i$.

*–Online–*

*-(On-1)* Let $x_i(t)$ be the $i$-th vehicle's state vector, find the smallest set index $l_i(t)$ containing $x_i(t)$, i.e.

$$l_i(t) := \min\{l \in \{0, 1, \ldots, N_i\} : x_i(t) \in \mathcal{T}_i^l(r_i(k_i))\} \tag{5}$$

*-(On-2)* If $l_i(t) = 0$ (i.e. $x_i(t) \in \mathcal{T}_i^0$) apply the control action given by terminal controller, otherwise apply the control action given by the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2021.3093280, IEEE Transactions on Automatic Control

solution of the following Quadratic Programming (QP) optimization problem:

$$\begin{aligned} u_i(t) = \arg\min_{u_i \in \mathcal{U}_i} ||A_i x_i(k) + B_i u_i - x^{eq}_{r_i}(k_i)||_2^2 \, s.t. \\ A_i x_i(t) + B_i u_i \in \tilde{\mathcal{T}}_i^{l_i(t)-1}(r_i(k_i)) \end{aligned} \tag{6}$$

***(ii) Waypoint update:*** Let's consider $r_i(k_i)$ as the current waypoint for the $i$-th UV and let's assume that the previously presented set-theoretic controller has terminal region and robust one-step controllable sets centered in $r_i(k_i)$, namely $\{\mathcal{T}_i^l(r_i(k_i))\}_{l=0}^{N_i}$. Let denote with $\bar{t} \in \mathbb{Z}_+$ the generic time instant when we want to switch to the successive waypoint $r_i(k_i+1)$. In principle, for linearity, we can switch waypoint and controller's DoA by simply re-centering the terminal region $\mathcal{T}_i^0(r_i(k_i))$ and the family of robust one-step controllable sets $\{\mathcal{T}_i^l(r_i(k_i)\}_{l=1}^{N_i}$ around the new equilibrium point $x^{eq}_{r_i(k_i+1)}$ associated to $r_i(k_i+1)$. Nevertheless, to guarantee that this operation is doable and preserves the vehicles' constraints (3), the following condition must be satisfied

$$x_i(\bar{t}) \in \bigcup_{l=0}^{N_i} \mathcal{T}_i^l(r_i(k_i+1)), \; \forall x_i(\bar{t}) \in \mathcal{T}_i^0(r_i(k_i)) \tag{7}$$

Indeed, condition (7) ensures that any state inside the current terminal region belongs to the $DoA_i$ of the controller shifted to the successive waypoint.

*Proposition 1: Let's consider the $i$-th vehicle model (1)-(3), the maximum distance $\delta_i$ between two successive waypoints $r_i(k_i)$ and $r_i(k_i+1)$, the terminal region $\mathcal{T}_i^0(0_2)$ centered in $x^{eq}_{0_2} = [0_2, 0^T_{n_i-2}]^T$, and a ball $\mathcal{B}_{\delta_i}(r_i(k_i+1)) \subset \mathbb{R}^n$ of radius $\delta_i$ centered in $x^{eq}_{r_i(k_i+1)} = [r_i(k_i+1)^T, 0^T_{n_i-2}]^T$. The $i$-th vehicle local controller with domain $DoA_i(r_i(k_i+1))$ guarantees condition (7) if the following set-inclusion holds true:*

$$DoA_i(r_i(k_i+1)) \supseteq \mathcal{B}_{\delta_i}(r_i(k_i+1)) \oplus \mathcal{T}_i^0(0_2) := \mathcal{W}_i \tag{8}$$

*with $\oplus$ denoting the Pontryagin/Minkowski set sum [23].*

*Proof* - For the considered $UV_i$ model, under zero inputs, the equilibrium states associated to $r_i(k_i)$ and $r_i(k_i+1)$ are $x^{eq}_{r_i(k_i)} = [r_i(k_i)^T, 0^T_{n_i-2}]^T$, and $x^{eq}_{r_i(k_i+1)} = [r_i(k_i+1)^T, 0^T_{n_i-2}]^T$, respectively. As a consequence, the distance in norm two between two successive equilibrium state vectors is at most $\delta_i$. To ensure safe waypoint switching from any state $x_i(t) \in \mathcal{T}_i^0(r_i(k_i))$ the $DoA_i(r_i(k_i+1))$ must be sufficiently large to cover the region $\mathcal{T}_i^0(r_i(k_i))$. This translates into the requirement that $DoA_i(r_i(k_i+1))$ must cover the region shaped by a ball of radius $\delta_i$ centered in $r_i(k_i+1)$ and enlarged by the dimension of the terminal region $\mathcal{T}_i^0$. Therefore, the condition $\forall x_i(\bar{t}) \in \mathcal{T}_i^0(r_i(k_i)) \rightarrow x_i(\bar{t}) \in DoA_i(r_i(k_i+1))$ is satisfied if the $DoA_i(r_i(k_i+1))$ is bigger or equal to the subset $\mathcal{W}_i \subset \mathbb{R}^n$ defined as the Minkowski sum between a ball centered in $r_i(k_i+1)$ and the terminal region associated to $0_2$, i.e., $\mathcal{W}_i := \mathcal{B}_{\delta_i}(r_i(k_i+1)) \oplus \mathcal{T}_i^0(0_2)$. □

***Remark** 2:* Please note that Proposition 1 provides a condition for the domain $DoA_i$ of the tracking controller so that (7) is satisfied and the waypoint update is feasible. Therefore, the condition (8) is used as the stopping criteria for the recursion (4).

*Proposition 2: The proposed set-theoretic MPC tracking controller enjoys the following properties: (i) For each waypoint $r_i(k_i)$, the vehicle trajectory will converge, in a finite number of steps (at most $N_i$) to the terminal region $\mathcal{T}_i^0(r_i(k_i))$ centered in the waypoints $r_i(k_i)$; (ii) For a constant waypoint $r_i$, $\mathcal{T}_i^0(r_i)$ defines an RCI region where the tracking error is UUB and constraints are fulfilled despite any admissible disturbance realization; (iii) The union of all the families of one-step controllable sets, i.e., $\bigcup_{\forall k_i} \{\mathcal{T}_i^l(r_i(k_i))\}_{l=0}^{N_i}$, defines a tube, passing by all the waypoints, which bounds all the admissible vehicle's trajectories.*

*Proof* - By collecting all the above developments and by resorting to the basic properties of the dual-mode set-theoretic MPC paradigm [19], [20].

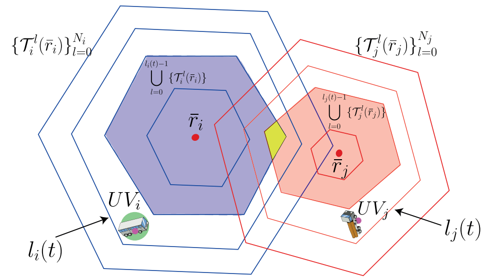


Fig. 2. Possibility of collision (yellow area) between $\boldsymbol{UV_i}$ and $\boldsymbol{UV_j}$, see condition (9).

## *B. Traffic Manager (TM) and Collision Avoidance*

In this subsection, the traffic manager logic and the collision avoidance strategy solving the control objective (**O2**) are presented. First, it is important to clarify which exchange of data is assumed between the local vehicles' controllers and the **TM**. This allows us to understand which information the **TM** can leverage to avoid collisions.

***Data exchange over time:***

- At $t = 0$: each $i$-th vehicle transmits to **TM** the computed family of robust one-step controllable sets centered in the first waypoint $r_i(0)$, $\{\mathcal{T}_i^l(r_i(0))\}_{l=0}^{N_i}$.
- At $t \geq 0$ :
  Each $i$-th vehicle sends to **TM**:
  - the current set-membership index $l_i(t)$ (see *(On-1)*)
  - the waypoint $\bar{r}_i$, where $\bar{r}_i = r_i(k_i+1)$ (next waypoint) if the $i$-th UV is in a terminal region, $\bar{r}_i = r_i(k_i)$ (current waypoint), otherwise. Hereafter, we denote with $\mathcal{I}_{sw}(t) \subseteq \mathcal{I}$, the set of UVs making request to switch waypoint at time $t$.

  **TM** sends to each $i$-th vehicle a binary variable which values are "Stop" or "Go".

The next proposition states the fundamental set-theoretic condition under which a collision between two UVs might happen. Please refer to Fig. 2 for a graphical illustration.

*Proposition 3: Let's consider two UVs, namely $UV_i$ and $UV_j$ modeled as (1)-(3). Let $\{\mathcal{T}_i^l(\bar{r}_i)\}_{l=0}^{N_i}$ and $\{\mathcal{T}_j^l(\bar{r}_j)\}_{l=0}^{N_j}$ be the families of one-step controllable sets currently used by the ST-MPC controllers to track the waypoints $\bar{r}_i$ and $\bar{r}_j$, respectively. If $l_i(t)$ and $l_j(t)$ are the set-membership indices at $t$, then a necessary condition for the existence of collisions at $t+1$ is:*

$$C_{\bar{r}_i \bar{r}_j}(l_i(t), l_j(t)) := \bigcup_{l=0}^{\max(l_i(t)-1,0)} \{\mathcal{T}_i^l(\bar{r}_i)\} \bigcap \bigcup_{l=0}^{\max(l_j(t)-1,0)} \{\mathcal{T}_j^l(\bar{r}_j)\} \neq \emptyset \tag{9}$$

*Proof* - According to the ST-MPC online algorithm (steps (*On-1*)-(*On-2*)), the one-step evolution of each vehicle is confined within a set which set-membership index is less or equal to the current one. In particular, if $UV_i$ is currently outside of the terminal region ($l_i(t) > 0$), then $l_i(t+1) < l_i(t)$, otherwise $l_i(t+1) = l_i(t) = 0$. The same arguments apply to $UV_j$. As a consequence, condition (9) represents a necessary but not sufficient condition for collisions at the next time instant.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2021.3093280, IEEE Transactions on Automatic Control

***Remark 3:*** If the waypoints $\bar{r}_i$ and $\bar{r}_j$ are kept constant, the vehicle's state trajectories $x_i(t)$ and $x_j(t)$ are UUB in $\bigcup_{l=0}^{l_i(t)}\{\mathcal{T}_i^l(\bar{r}_i)\}$ and $\bigcup_{l=0}^{l_j(t)}\{\mathcal{T}_j^l(\bar{r}_j)\}$, respectively. As a consequence, if no collisions are predicted at $\bar{t}$, then no collisions can occur for any $t > \bar{t}$.

Given the result in *Proposition* 3, we want to design a conservative but effective traffic manager capable of ensuring that, for any pair $(i, j)$ of UVs in $\mathcal{I}$, the potential collision condition (9) is never reached. To this end, first, a connectivity graph modeling all the possible intersections is built, and then a collision avoidance strategy capable of minimizing the number of vehicles to be stopped is illustrated.

For feasibility reasons, we assume that at $t = 0$ the condition (9), $\forall$ pair $(i, j)$ in $\mathcal{I}$, is not satisfied (no initial collisions). Therefore, until no vehicle makes a request to update the current waypoint, namely $r_i(k_i)$, $\forall\, i \in \mathcal{I}$ , no collisions are possible (see *Remark* 3). As soon as the first vehicle makes the request to switch waypoint (i.e. the vehicle has reached the current terminal region), an undirected connectivity graph [26] $\mathcal{G}(t) := (\mathcal{V}, \mathcal{E}(t))$ is built with $\mathcal{V} = \mathcal{I}$ the vehicles set and $\mathcal{E}(t)$ the edge set modeling potential collisions. Each edge $e_{ij}(t) \in \mathcal{E},\ i, j \in \mathcal{V}$, is computed as follows:

$$e_{ij}(t) = \begin{cases} 1 & \text{if} \quad i \in \mathcal{I}_{sw} \text{ and } \mathcal{C}_{\bar{r}_i\bar{r}_j}(l_i(t), l_j(t)) \neq \emptyset \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

where $\bar{r}_p = r_p(k_p+1)$ if $p \in \mathcal{I}_{sw}$, $\bar{r}_p = r_p(k_p)$ otherwise, $p = i, j$. If at the time $t$, $\exists\, e_{ij}(t) \neq 0$, a collision avoidance strategy must be activated. In the sequel, we denote with $\mathcal{I}_{stop} \subseteq \mathcal{I}$ the set of vehicles that need to be stopped. It is important to remark that all the $UV_is$ making a switching request, $i \in \mathcal{I}_{sw}$, are already within RCI regions, i.e. $x_i(t) \in \mathcal{T}_i^0(r_i(k_i))$. Therefore, if needed, they can be stopped for an indefinitely long time period without causing collisions (see *Remark* 3). Moreover, $UV_i$ not making any request of waypoint switch cannot be stopped (they are not in the terminal RCI region). In order to stop the minimum number of UVs, the following procedure is applied:

**(*P1*)** Find the vehicle $\bar{i} \in \mathcal{I}_{sw}$ with the highest adjacency degree

$$\bar{i} = \arg\max_{i \in \mathcal{I}_{sw}} \Delta[\mathcal{G}] \tag{11}$$

with $\Delta[\mathcal{G}]$ the diagonal degree matrix containing the adjacency degrees for all vehicles [26]. If more than one vehicle has the same highest adjacency degree, we assume that the above function randomly selects one of them (vehicles have no priorities).

**(*P2*)** Add $\bar{i}$ to $\mathcal{I}_{stop}$ and set to zero all the edges connected to the node $\bar{i}$, i.e. $e_{\bar{i}j} = 0\ \forall\, j \in \mathcal{I}$

**(*P3*)** If $\exists\, e_{ij}(t) \in \mathcal{E}(t) :\ e_{ij}(t) \neq 0$, goto Step (P1)

***Remark 4:*** Please note that the procedure (***P1***)-(***P3***) terminates when a disconnected graph $\mathcal{G}(t)$ is obtained. Therefore, in the worst-case scenario at least one vehicle will be always allowed to move to the next waypoint, so avoiding deadlocks.

### C. Computational Algorithms

All the above developments can be collected in the following computational algorithms:

Traffic Manager **(TM)**

1: **if** $\mathcal{I}_{sw} = \emptyset$ **then** $\mathcal{I}_{stop} = \emptyset$
2: **else** Build $\mathcal{G}(t) = (\mathcal{V}(t), \mathcal{E}(t))$ as in (10)
3:     **if** $\exists\, e_{ij}(t) \neq 0$ **then** Compute $\mathcal{I}_{stop}$ by (P1)-(P3)
4:     **else** $\mathcal{I}_{stop} = \emptyset$
5:     **end if**
6: **end if**
7: $\forall\, i \in \mathcal{I}$, if $i \in \mathcal{I}_{stop}$, send "Stop", otherwise "Go"

Set-Theoretic MPC $i$ **(ST-MPC-i)**

1: Use (5) to find the smallest set index $l_i(t)$ containing $x_i(t)$
2: **if** $l_i(t) == 0$ **then** $\bar{r}_i \leftarrow r_i(k_i+1)$
3: **else** $\bar{r}_i \leftarrow r_i(k_i)$
4: **end if**
5: Send to **TM**: $l_i(t)$, $\bar{r}_i$
6: **if** $l_i(t) == 0$ & **TM**=="Go" **then**
7:     $r_i(k_i) \leftarrow r_i(k_i+1)$, $k_i \leftarrow k_i + 1$;
8:     Update $l_i(t)$ by using (5)
9: **end if**
10: **if** $l_i(t) == 0$ **then** $u_i(t) = K_i^0(x_i(t) - x_{r_i}^{eq}(k_i))$
11: **else** Find $u_i(t)$ by solving opt. (6)
12: **end if**

***Remark 5:*** The computational complexity of the proposed **TM** algorithm is mainly related to the construction of $\mathcal{G}(t)$ in Step 2. In particular, to test if between two vehicles $i$ and $j$ there is a collision possibility (i.e., $e_{ij}(t) = 1$), then a set-membership test must be performed. Assuming a polyhedral representation for the robust one-step controllable sets $\mathcal{T}_i^l$, each test requires the solution of a simple linear programming (LP) optimization problem solvable in polynomial time. Therefore, to completely build $\mathcal{G}(t)$, the number of LP problems that must be solved is equal to $\sum_{\eta=1}^{|\mathcal{I}_{sw}|}(S-\eta)$ where $|\mathcal{I}_{sw}|$ is the number of vehicles making a waypoint switch request at the time $t$ and $S$ is the total number of vehicles. On the other hand, the local **ST-MPC-i** is mainly related to the solution of the QP optimization problem defined in (6). Therefore, contrary to the existing DMPC solutions, the proposed approach does not requires inter-vehicles communications and non-convex optimizations. As an example, at each iteration, the distributed approach in [16] requires the computation of $S$ MILPs while the proposed solution requires $S(S-1)/2$ LPs (worst-case) and 1 QP per vehicle. Finally, by noticing that the required LP problems are independent of each other their computation can be parallelized. As a consequence, by resorting to a parallel computing platform, the time required by the TM operations can be drastically decreased, so making TM affordable in practical scenarios.

*Proposition 4: Let's consider a set $\mathcal{I}$ of UVs modeled as in (1)-(3) where each $i$-th UV is equipped with the **ST-MPC-i** local controller and the vehicles waypoint switches are coordinated by a centralized traffic manager which logic is described by the **TM** algorithm. If the UVs start from a feasible collision-free initial condition, i.e.*

$$\exists\, l_i(0) \geq 0 :\ x_i(0) \in \bigcup_{l=0}^{l_i(0)} \mathcal{T}_i^l(r_i(0)),\ \ \forall i \in \mathcal{I}$$
$$\bigcup_{l=0}^{l_i(0)} \mathcal{T}_i^l(r_i(0)) \cap \bigcup_{l=0}^{l_j(0)} \mathcal{T}_j^l(r_j(0)) = \emptyset,\ \forall (i,j),\ i \neq j,\ i,j \in \mathcal{I} \tag{12}$$

*then, **TM** guarantees the absence of collisions, i.e.* $||p_i(t) - p_j(t)||_2 > 0,\ \ \forall t \in \mathbb{Z}_+,\ \forall i, j \in \mathcal{I},\ i \neq j$ *regardless of the UVs trajectories* $\boldsymbol{r}_i$, $\forall\, i \in \mathcal{I}$.

*Proof* - If each $i$-th UV starts from a feasible collision-free initial condition, see (12), then **ST-MPC-i** controller is capable of steering each UV state trajectory within the current terminal RCI region in a finite number of steps (see ST-MPC-i, Steps 10-11) where it can be confined for an arbitrarily long time interval. Moreover, according to the results in *Proposition* 3 and *Remark* 3, until no waypoint switch occurs, collisions among vehicles are not possible. In *Proposition* 1 it has been proved that starting from a terminal region,

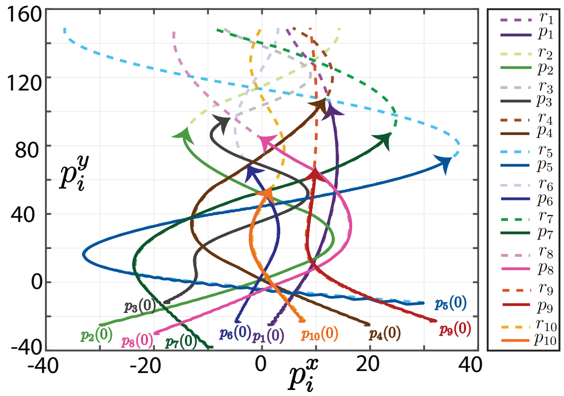


Fig. 3. $\boldsymbol{UV}$ Waypoints and trajectories for $\boldsymbol{t \in [0,\, 100]s}$.

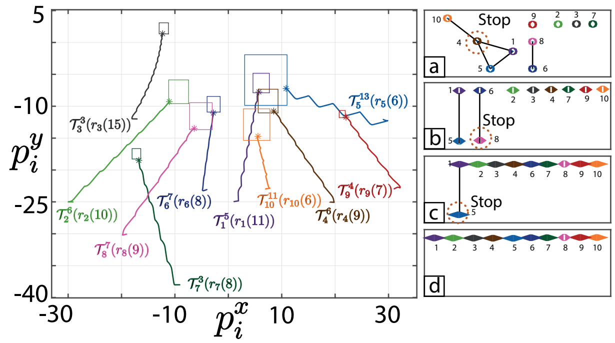


Fig. 4. Potential collisions at $\boldsymbol{t = 8.9s}$ and connectivity graphs.

e.g. $\mathcal{T}_i^0(r_i(k_i))$ waypoint switches $r_i(k_i) \rightarrow r_i(k_i+1)$ are always feasible and preserve vehicle constraints (3). On the other hand, to avoid collisions among the vehicles, switches can be accomplished only if the collision avoidance condition (9) is preserved between any pair of vehicles. To this end, each vehicle, before switching, asks permission to the **TM** (see ST-MPC-i, Step 6) who collects all the requests. The **TM**, by building the connectivity graph (see TM, Step 2), checks if any of the requested waypoint switches do not preserve (9) (see TM, Step 3). If collisions are detected, then the procedure (P1)-(P3) is activated (see TM, Step. 3) to deny the waypoint switch to the minimum number of vehicles. The latter is sufficient to maintain the vehicles controllers' domain mutually disjointed, i.e. $\bigcup_{l=0}^{l_i(t)} \mathcal{T}_i^l(r_i(k_i))) \cap \bigcup_{l=0}^{l_j(t)} \mathcal{T}_j^l(r_j(k_j)) = \emptyset,\ \forall (i,j),\ i \neq j,\ i,j \in \mathcal{I}$, and ensure absence of collisions.

## IV. Simulation

In this section, the effectiveness of the proposed control architecture is testified by means of a simulation example involving ten $\mathcal{I} = \{1, \ldots, 10\}$ UVs. The UVs dynamics are described by mean of a double integrator model [16] subject to bounded state and input constraints and disturbances. The model has been discretized using a sampling time $T_s = 0.1\,\text{sec}$. The state space vector $x_i \in \mathbb{R}^4$ includes the vehicle's 2D coordinates $p_i = [p_i^x,\, p_i^y]^T$ and velocities $z_i = [v_i^x,\, v_i^y]^T$, while $u_i = [a_i^x, a_i^y] \in \mathbb{R}^2$ is the acceleration vector. The disturbance set and the constraints are the following

$$|v_i^x| = |v_i^y| \leq \bar{v}_i,\ |u_i^x| = |u_i^y| \leq \bar{u}_i,$$
$$\mathcal{D}_i = \{d = [d_1, \ldots, d_4]^T : |d_j| \leq \bar{d}_i,\ j = 1, \ldots, 4\}$$

where for $i \in \{1,2,3\} \rightarrow \bar{v}_i = 20,\ \bar{d}_i = 0.06,\ i \in \{4,5\} \rightarrow \bar{v}_i = 25,\ \bar{d}_i = 0.085,\ i \in \{6,9,10\} \rightarrow \bar{v}_i = 8,\ \bar{d}_i = 0.07,\ i \in \{7,8\} \rightarrow \bar{v}_i = 18,\ \bar{d}_i = 0.065,\ \forall i \rightarrow \bar{u}_i = 4$. Each vehicle's reference generator provides a sequence of waypoints $r_i(k_i)$ as shown in Fig. 3. The maximum distance $\delta_i,\ \forall\, i \in \mathcal{I}$ between two successive waypoints are: $i \in \{1,2,3\} \rightarrow \delta_i = 4.02,\ i \in \{4,5\} \rightarrow \delta_i = 5.62,\ i \in \{6,9,10\} \rightarrow \delta_i = 3.22,\ i \in \{7,8\} \rightarrow \delta_i = 3.91$. According to the proposed ST-MPC strategy, for each vehicle, a terminal controller and an RCI set have been offline computed as in [22]. Moreover, to assure that the vehicles' controller domains satisfy the waypoint switching feasibility condition (7), a family of $N_i$ robust controllable sets $\{\mathcal{T}_i^l\}_{l=1}^{N_i}$ has been computed for each vehicle, where for $i \in \{1,2,3\} \rightarrow N_i = 21,\ i \in \{4,5\} \rightarrow N_i = 19,\ i \in \{6,9,10\} \rightarrow N_i = 15,\ i \in \{7,8\} \rightarrow N_i = 18$. By considering the vehicle's initial positions as shown in Fig. 3, the obtained simulation results are collected in Figs. 3-5. In Fig. 3, the vehicle's trajectories are depicted for the time interval $[0,\, 100]s$. The trajectories show how the **ST-MPC-i** controllers are able to track the switching waypoints in Fig. 3 despite constraints and disturbances. Moreover, it is possible to notice that the obtained paths have potential collision points. Therefore, it is worth to investigate how the traffic manager actions are essential to avoid collisions. To better understand the **TM** *modus operandi*, we shall refer to Fig. 4 where the simulation has been paused at $t = 8.9s$. Specifically, at the considered screenshot, the vehicle's set membership scenario is the following: $x_1 \in \mathcal{T}_1^0(r_1(10)),\ x_2 \in \mathcal{T}_2^0(r_2(9)),\ x_3 \in \mathcal{T}_3^0(r_3(14)),\ x_4 \in \mathcal{T}_4^0(r_4(8)),\ x_5 \in \mathcal{T}_5^0(r_5(5)),\ x_6 \in \mathcal{T}_6^8(r_6(8)),\ x_7 \in \mathcal{T}_7^4(r_7(8)),\ \ x_8 \in \mathcal{T}_8^0(r_8(8)),\ x_9 \in \mathcal{T}_9^5(r_9(7)),\ x_{10} \in \mathcal{T}_{10}^0(r_{10}(5))$. The **TM**, first collects all the waypoint switch requests and set-membership indices. At $t = 8.9s$, the set of UVs in a terminal region making a waypoint switch request is $\mathcal{I}_{sw} = \{1,2,3,4,5,8,9\}$. Given the collected information, the **TM** builds the connectivity graph $\mathcal{G}(8.9)$ according to (10). In Fig. 4, the current vehicles' positions are shown with a star symbol and the rectangular areas (matched by color) represent the regions where the one-step evolution (at $t = 9s$) of each agent will be confined. By construction, according to the waypoint switch feasibility condition (7), the vehicles in the terminal region also belong to the family of the one-step controllable set associated with the successive waypoint. In particular, at $t = 8.9s$: $x_1 \in \mathcal{T}_1^5(r_1(11)),\ x_2 \in \mathcal{T}_2^6(r_2(10)),\ x_3 \in \mathcal{T}_3^3(r_3(15)),\ x_4 \in \mathcal{T}_4^6(r_4(9)),\ x_5 \in \mathcal{T}_5^{13}(r_5(6)),\ x_8 \in \mathcal{T}_8^7(r_5(9)),\ x_{10} \in \mathcal{T}_{10}^{11}(r_{10}(6))$. Since the constructed families of one-step controllable sets are nested, in Fig. 4, we show only the outer sets. The connectivity graph in Fig. 4.a summarizes all the possible collisions (9) at $t = 8.9s$. In particular, the graph presents potential collisions among the vehicles $1-4$, $4-5$, $1-5$, $4-10$ and $6-8$. Therefore, according to the **TM** algorithm (see its step 3) the (P1)-(P3) procedure is activated to avoid collisions by stopping the minimum number of vehicles among the ones making a waypoint switch request. In the first iteration, the $UV_4$ (the node with the highest degree) is stopped and added to $\mathcal{I}_{stop}$. As a consequence, all the edges connected to $UV_4$ are also removed. The resulting connectivity graph and remaining intersections are shown in Fig. 4.b. Since collisions between the vehicles $1-5$ and $6-8$ are still possible, a second iteration of (P1)-(P3) is executed and the vehicle $UV_8$ is added to $\mathcal{I}_{stop}$. Fig. 4.c. shows the resulting graph with a single collision possibility between $UV_1$ and $UV_5$. Therefore, since both vehicles have the same connectivity degree either $UV_1$ or $UV_5$ could be stopped. In the simulation, $UV_5$ is added to $\mathcal{I}_{stop}$, i.e. $\mathcal{I}_{stop} = \{4,5,8\}$. The completely disconnected in Fig. 4.d results where no collisions are possible. Therefore, the **TM** operations are concluded: $UV_4, UV_5$ and $UV_8$ are stopped while $UV_1, UV_2, UV_3$, and $UV_{10}$ are allowed to switch waypoint. In Fig. 5, the vehicles set-membership index signal is shown for $UV_3, UV_5$ and $UV_9$ in the time interval $[0,20]s$ (the time interval has been shortened to improve the figure's readability). Such signal allows us to better clarify the STOP and GO commands received by these vehicles according to the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAC.2021.3093280, IEEE Transactions on Automatic Control

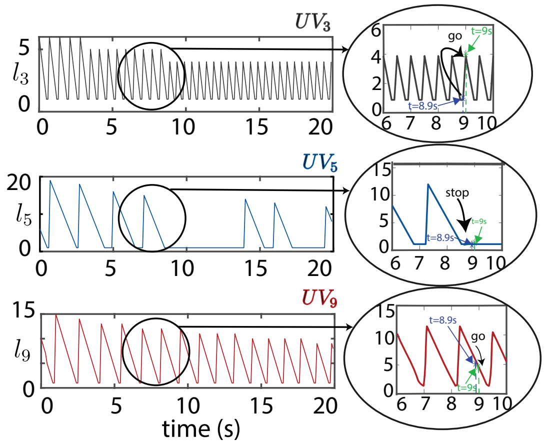


Fig. 5. Vehicles' set membership indices in the time interval $[\mathbf{0}-\mathbf{20}]\boldsymbol{s}$.

**TM** operations previously described. In the absence of collisions, the signals $l_i(t)$, by construction, have a reverse sawtooth shape. The wave ramps downward while the vehicles move within the family of computed one-step controllable sets, and sharply rises when a waypoint switch occurs. On the other hand, when the signal $l_i(t)$ holds constant for more than one sampling time, then it means that the vehicle $i$ has received a STOP command. It is worth noticing in Fig. 5 that, according to the developed theory, a STOP signal can be received only by the vehicles making switch request, i.e. from the vehicles within a terminal region ($l_i(t) = 0$). As an example, in the previously described potential collision happening at $t = 8.9s$, the **TM** imposes a STOP on three of the seven vehicles making a switch request, i.e. $UV_4, UV_5$ and $UV_8$. As a consequence, in the zoom-in subplot in Fig. 5, it is possible to appreciate what follows: the signal $l_5(8.9)$ stays constant to zero, meaning that the waypoint switch has been denied for $UV_5$; the index $l_3(8.9)$ jumps from 0 to $l_3(9) = 11$, testifying that the waypoint switch has been granted to $UV_3$; the signal $l_9(8.9)$ keeps decreasing showing that $UV_9$ moves closer to the current waypoint (vehicles that do not belong to the terminal region won't be stopped). Finally, for the interested reader, the demo of the performed simulation for 20 vehicles is available at the following weblink: `https://tinyurl.com/34n5c3yt`.

## V. Conclusions

In this paper, we have presented a novel solution to deal with the collision avoidance problem for heterogeneous constrained vehicles moving in a shared environment. We have proposed a control architecture where each vehicle is equipped with a local set-theoretic MPC tracking controller and a centralized traffic manager, exploiting simple set-membership arguments, guarantees absence of collisions. We have proved that the proposed solution is sufficient to guarantee collision avoidance despite vehicle constraints, disturbance realization and desired vehicle's trajectories. Moreover, by modeling the potential collisions as a connectivity graph, we have proposed a strategy that minimizes, at each time, the number of vehicles that need to be stopped. Future works will focus on extending the proposed control architecture to deal with nonlinear vehicle models and to reduce the conservativeness of the collision avoidance check performed by the traffic manager.

## References


[1] B. Paden, M. Čáp, S. Z. Yong, and E. Yershov, D.and Frazzoli, "A survey of motion planning and control techniques for self-driving urban vehicles," *IEEE Trans. on Intelligent Vehicles*, vol. 1, no. 1, pp. 33–55, 2016.

[2] J. Wit, C. D. Crane III, and D. Armstrong, "Autonomous ground vehicle path tracking," *Journal of Robotic Systems*, vol. 21, no. 8, pp. 439–449, 2004.

[3] A. P. Aguiar and J. P. Hespanha, "Trajectory-tracking and path-following of underactuated autonomous vehicles with parametric modeling uncertainty," *IEEE Trans. on Automatic Control*, vol. 52, no. 8, pp. 1362–1379, 2007.

[4] M. Egerstedt and X. Hu, "Formation constrained multi-agent control," *IEEE trans. on robotics and automation*, vol. 17, no. 6, pp. 947–951, 2001.

[5] S. Mastellone, D. M. Stipanović, C. R. Graunke, K. A. Intlekofer, and M. W. Spong, "Formation control and collision avoidance for multi-agent non-holonomic systems: Theory and experiments," *Int. Journal of Robotics Research*, vol. 27, no. 1, pp. 107–126, 2008.

[6] J. Minguez, F. Lamiraux, and J.-P. Laumond, "Motion planning and obstacle avoidance," in *Springer handbook of robotics*. Springer, 2008, pp. 827–852.

[7] X. Dong, B. Yu, Z. Shi, and Y. Zhong, "Time-varying formation control for unmanned aerial vehicles: Theories and applications," *IEEE Trans. on Control Systems Technology*, vol. 23, no. 1, pp. 340–348, 2015.

[8] J. A. Fax and R. M. Murray, "Information flow and cooperative control of vehicle formations," *IEEE Trans. on Automatic Control*, vol. 49, no. 9, pp. 1465–1476, 2004.

[9] H. Liu, X. Dong, F. L. Lewis, Y. Wan, and K. P. Valavanis, "Robust formation control for multiple quadrotors subject to nonlinear dynamics and disturbances," in *Int. Conf. on Control and Automation*, 2018, pp. 58–62.

[10] T. Mylvaganam, M. Sassano, and A. Astolfi, "A differential game approach to multi-agent collision avoidance," *IEEE Trans. on Automatic Control*, vol. 62, no. 8, pp. 4229–4235, 2017.

[11] A. Colombo and D. Del Vecchio, "Least restrictive supervisors for intersection collision avoidance: A scheduling approach," *IEEE Trans. on Automatic Control*, vol. 60, no. 6, pp. 1515–1527, 2014.

[12] A. Mondal, C. Bhowmick, L. Behera, and M. Jamshidi, "Trajectory tracking by multiple agents in formation with collision avoidance and connectivity assurance," *IEEE Systems Journal*, vol. 12, no. 3, pp. 2449–2460, 2017.

[13] L. Wang, A. D. Ames, and M. Egerstedt, "Safety barrier certificates for collisions-free multirobot systems," *IEEE Trans. on Robotics*, vol. 33, no. 3, pp. 661–674, 2017.

[14] X. Wang, M. Kloetzer, C. Mahulea, and M. Silva, "Collision avoidance of mobile robots by using initial time delays," in *IEEE Conf. on Decision and Control*, 2015, pp. 324–329.

[15] M. Chen, J. C. Shih, and C. J. Tomlin, "Multi-vehicle collision avoidance via hamilton-jacobi reachability and mixed integer programming," in *IEEE Conf. on Decision and Control*, 2016, pp. 1695–1700.

[16] A. Richards and J. How, "Decentralized model predictive control of cooperating uavs," in *IEEE Conf. on Decision and Control*, vol. 4, 2004, pp. 4286–4291.

[17] K.-D. Kim and P. R. Kumar, "An mpc-based approach to provable system-wide safety and liveness of autonomous ground traffic," *IEEE Trans. on Automatic Control*, vol. 59, no. 12, pp. 3341–3356, 2014.

[18] D. P. Bertsekas and I. B. Rhodes, "On the minimax reachability of target sets and target tubes," *Automatica*, vol. 7, no. 2, pp. 233–247, 1971.

[19] F. Blanchini and S. Miani, *Set-theoretic methods in control*. Springer, 2008.

[20] D. Angeli, A. Casavola, G. Franzè, and E. Mosca, "An ellipsoidal off-line mpc scheme for uncertain polytopic discrete-time systems," *Automatica*, vol. 44, no. 12, pp. 3113–3119, 2008.

[21] J.-P. Hubaux, S. Capkun, and J. Luo, "The security and privacy of smart vehicles," *IEEE Security & Privacy*, vol. 2, no. 3, pp. 49–55, 2004.

[22] M. Bagherzadeh and W. Lucia, "Multi-vehicle reference tracking with guaranteed collision avoidance," in *European Control Conf.*, 2019, pp. 1574–1579.

[23] F. Borrelli, A. Bemporad, and M. Morari, *Predictive control for linear and hybrid systems*. Cambridge University Press, 2017.

[24] W. Lucia, D. Famularo, and G. Franzè, "A set-theoretic reconfiguration feedback control scheme against simultaneous stuck actuators," *IEEE Trans. on Automatic Control*, vol. 63, no. 8, pp. 2558–2565, 2017.

[25] S. V. Rakovic, E. C. Kerrigan, K. I. Kouramas, and D. Q. Mayne, "Invariant approximations of the minimal robust positively invariant set," *IEEE Trans. on Automatic Control*, vol. 50, no. 3, pp. 406–410, 2005.

[26] Z. Lin, B. Francis, and M. Maggiore, "Necessary and sufficient graphical conditions for formation control of unicycles," *IEEE Trans. on Automatic Control*, vol. 50, no. 1, pp. 121–127, 2005.