Set-theoretic Time-based Trajectory Synchronization Approach for Skid-Steered Robotic units subject to Constraints, Uncertainties, and External Disturbances

Alessia Ferraro[1], Claudio De Capua[2] and Valerio Scordamaglia[3]

***Abstract*— This paper presents a set-theoretic time-based trajectory synchronization method to coordinate the skid-steered units subject to slippage phenomena of a multi-robot system. The closed-loop trajectory tracking error dynamics are expressed in the form of an uncertain system subject to external disturbances and actuation constraints. Collision avoidance and coordination are achieved through synchronization of robot trajectories by imposing delays on platform departures. The procedure exploits the trajectory's feasibility property by solving optimization problems involving Linear Matrix Inequalities (LMIs) constraints. The algorithm operates in separate offline and online phases. In the offline phase, the time delay intervals useful for synchronization are calculated based on the planned feasible trajectories. Then, in the online phase, the actual delay is calculated by solving an optimization problem that minimizes the occupancy time of the shared operational space. The results of several numerical simulations are presented and discussed to demonstrate the proposed solution's effectiveness.**

***Index Terms*— collision avoidance, coordination, feasible trajectory planning, multi-robot systems, temporal scheduling;**

I. Introduction

The use of multi-robot systems (MRSs) in performing complex tasks is increasingly common in modern technology applications, due to their collaborative capabilities that overcome the limitations of a single robot [1]. These applications leverage the collective power and versatility of MRSs to manage tasks that require coordination and efficiency. However, significant challenges still remain, among others, the problem of planning trajectories that can be followed by robots even under uncertainty and external disturbances in known or partially known environments with obstacles [2], [3], the challenge of accurate trajectory tracking even in the presence of actuation constrains [4], and the problem of coordinating the movements of multiple robots within a shared space [5] are key areas of active research. The uncertainties can arise from a variety of factors, such as environmental conditions (e.g., variation in ground friction), control loop delays, or even actuation constraints.

There are several papers in the literature that address the problem of coordinating multiple robots [6]–[11]. In [12], the authors propose an approach that uses a velocity profile assignment solution to solve the problem of coordination. The study focuses on the collision-free coordination of multiple robots with kinodynamic constraints along specified paths. The solution generates continuous velocity profiles for multiple robots that satisfy dynamics constraints. The approach combines techniques from optimal control and mathematical programming. A strategy for supervising the evolving movement of mobile robots in the same environment is presented in [13]. The solution involves building a specific Petri-net model for available trajectories, and using resource allocation techniques based on capacity-constrained regions and deadlock-free execution.

Other methods address the problem of movement coordination and collision avoidance through platform prioritization, and allocation of delays. In [14] the authors develop a method that uses k-means clustering to balance tasks between robots and a collision model with time priority constraints. This model defines interference zones between the robots. To avoid collisions, a directed acyclic graph (DAG) is constructed to represent the priority constraints. The final scheduling creates a schedule that minimizes the completion time, ensuring that robots do not collide. In [15], the collision avoidance problem is addressed for a multi-robot system through the impositions of delays. Specifically, the trajectories consist of sequences of regions to be followed, and a known travel time for each region. The problem of avoiding collisions between robots is solved by imposing initial delays for each trajectory. Some contributions propose solutions that address the problem of coordinating multi-agent systems during the trajectory planning phase, which are effective as they do not require the installation of dedicated hardware for the collision management task.

In this paper, the coordination and collision avoidance for the skid-steered units of an MRS are achieved using a set-theoretic time-based trajectory synchronization method. The solution exploits the feasibility properties of the planned trajectories, which ensure robustness to model uncertainties, external disturbances, and actuation constraints. By delaying the departure times of the robots, the solution guarantees a minimum safety distance between the platforms during trajectory tracking. The approach comprises two phases: an offline phase, in which the intervals of the synchronization delay are identified by solving a minimization problem with linear matrix inequalities (LMIs) constraints, and an online phase, in which the actual robot departure delays

*This work was supported by the research project - ID:P2022XER7W "CHEMSYS: Cooperative Heterogeneous Multi-drone SYStem for disaster prevention and first response" granted by the Italian Ministry of University and Research (MUR) within the PRIN 2022 PNRR program, funded by the European Union through the PNRR program.

A. Ferraro, C. De Capua and V. Scordamaglia are with the Department DIIES, of the University "Mediterranea" of Reggio Calabria, 89100, Italy, e-mail: alessia.ferraro@unirc.it; claudio.decapua@unirc.it; valerio.scordamaglia@unirc.it;

are computed by solving a linear optimization problem that minimizes the operational environment occupation time.

The paper is structured as follows. Sect. II presents the uncertain closed-loop dynamics of the trajectory tracking error for a skid-steered robotic platform subjected to external disturbances. Sect. III proposes a procedure to calculate the synchronization delay intervals, and the actual delays for the robot starts, through solving set-theoretic optimization problems. Finally, Sect. IV presents the results of several numerical simulations which consider multiple skid-steered mobile robots operating in an indoor environment to demonstrate the effectiveness of the proposed solution.

II. Skid-steered tracked mobile robot mathematical model

Consider a set of autonomous skid-steered mobile robots operating in a shared environment. Each robot, treated as a rigid body moving on a horizontal plane, has its pose at time t given by the vector $q(t) = \begin{bmatrix} x(t) & y(t) & \theta(t) \end{bmatrix}^T$, where x and y are the coordinates of robot's position, and θ is the orientation in a global reference frame $\mathbf{E}$. The motion of the robot is controlled by the velocities vector $u(t) = \begin{bmatrix} v(t) & \omega(t) \end{bmatrix}^T$ including the forward velocity v and the rotational velocity ω. Thus the robot's first-order kinematic model can be described as follows

$$\dot{q}(t) = \begin{cases} \dot{x} = cos(\theta(t))\, v(t), \\ \dot{y} = sin(\theta(t))\, v(t), \\ \dot{\theta} = \omega(t). \end{cases} \tag{1}$$

Let $\hat{u}(t) = \begin{bmatrix} \hat{v}(t) & \hat{\omega}(t) \end{bmatrix}^T$ denote the desired control variables, with $\hat{v}$ and $\hat{\omega}$ as the desired forward and rotational velocities. Due to differential driving, tracked robots are subjected to slip and skid phenomena, leading to $v \neq \hat{v}$ and/or $\omega \neq \hat{\omega}$. Following [16], we model the track-soil interaction with time-varying sliding coefficients $\mu_r(t)$ and $\mu_l(t)$ for the right and left tracks, and relate the motor angular velocities $\rho(t) = \begin{bmatrix} \rho_r(t) & \rho_l(t) \end{bmatrix}^T$ to the robot's forward and rotational velocities through the following relationship:

$$u(t) = \begin{bmatrix} R/2 & R/2 \\ R/G & -R/G \end{bmatrix} \begin{bmatrix} \mu_r(t) & 0 \\ 0 & \mu_l(t) \end{bmatrix} \begin{bmatrix} \rho_r(t) \\ \rho_l(t) \end{bmatrix}, \tag{2}$$

where R is the radius of the gears connecting tracks and motors, and G is the distance between the tracks. The motor velocities required to achieve the desired control velocities are calculated assuming unit sliding coefficients

$$\rho(t) = \begin{bmatrix} R/2 & R/2 \\ R/G & -R/G \end{bmatrix}^{-1} \hat{u}(t). \tag{3}$$

By recombining the equations, the kinematic model (1) can be rewritten in the following form:

$$\dot{q}(t) = \begin{bmatrix} cos(\theta(t)) & 0 \\ sin(\theta(t)) & 0 \\ 0 & 1 \end{bmatrix} \cdot u(t) \tag{4}$$

where

$$u(t) = \begin{bmatrix} R/2 & R/2 \\ R/G & -R/G \end{bmatrix} \begin{bmatrix} \mu_r(t) & 0 \\ 0 & \mu_l(t) \end{bmatrix} \begin{bmatrix} R/2 & R/2 \\ R/G & -R/G \end{bmatrix}^{-1} \begin{bmatrix} \hat{v}(t) \\ \hat{\omega}(t) \end{bmatrix}. \tag{5}$$

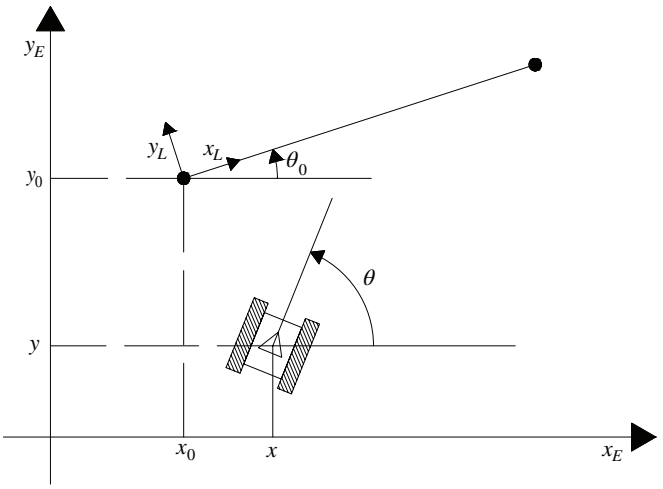

Fig. 1. Reference frames. The pose of the robot in the $\mathbf{E}$ frame is denoted as $\{x\ y\ \theta\}$, and the reference frame $\mathbf{L}$ for trajectory tracking has its origin at the first point of the segment, with the x-axis oriented along it.

A. Trajectory tracking error

We define $q_D(t) = \begin{bmatrix} v_D \cdot t & 0 & 0 \end{bmatrix}^T$ as the desired pose of the robot along the trajectory's segments at a time t, in a local reference frame $\mathbf{L}$ centered on $\{x_0\ y_0\}$, and rotated with respect to $\mathbf{E}$ by an angle θ_0, see Fig. 1. The actual pose of the robot in $\mathbf{L}$ is expressed by the following roto-translation:

$$q_L(t) = R_E^L(\theta_0)(q(t) - q_0) \tag{6}$$

where $q_0 = \begin{bmatrix} x_0 & y_0 & \theta_0 \end{bmatrix}^T$, and $R_E^L(\theta_0)$ is given by:

$$R_E^L(\theta_0) = \begin{bmatrix} cos(\theta_0) & sin(\theta_0) & 0 \\ -sin(\theta_0) & cos(\theta_0) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{7}$$

Let $e(t) = \begin{bmatrix} e_x(t) & e_y(t) & e_\theta(t) \end{bmatrix}^T = q_L(t) - q_D(t)$ be the trajectory tracking error at time t, expressed in $\mathbf{L}$. By applying classical linearization arguments, the following linear time-invariant mathematical representation is defined

$$\dot{e}(t) = Ae(t) + B\delta u(t) + B_d d(t). \tag{8}$$

This describes the trajectory tracking error in the nominal condition $\nabla(t) = \{q_D(t), u_D, \mu_r^N, \mu_l^N\}$, where $u_D = \begin{bmatrix} v_D & 0 \end{bmatrix}^T$ is the vector of desired robot velocities, $\mu_r^N = \mu_l^N = 1$ are the nominal sliding coefficients, and where $\delta_u = \begin{bmatrix} \hat{v} - v_D & \hat{\omega} - \omega_D \end{bmatrix}^T$, and $d = \begin{bmatrix} \mu_l - \mu_l^N & \mu_r - \mu_r^N \end{bmatrix}^T$ represents an external disturbance acting on the system. Assume that the multi-robot system has a fully decentralized architecture [17], in which each robot is independent and has its controller for trajectory tracking. Assume that each robot has no reactive capabilities implemented [18], and no robots exchange information with other robots in the system. The dynamics describing the closed-loop trajectory tracking error for a single robot following a straight trajectory with constant velocity can be expressed by an uncertain discrete linear time-invariant system with norm-bounded uncertainty

[19] subject to an external disturbance as follows

$$\xi_{k+1} = \phi\xi_k + H_d d_k + B_p p_k, \tag{9}$$

$$p_k = \Delta_k q_k, \tag{10}$$

$$q_k = \Sigma_q \xi_k, \tag{11}$$

where $\|\Delta_k\| < 1 \; \forall k \geq 0$, The interested reader can find further mathematical details in [16]. Assume that admissible state and control input variations are constrained according to the following ellipsoidal sets:

$$\xi \in \Omega_\xi,\, \Omega_\xi = \{\xi \in \mathcal{R}^{n_\xi} : \xi^T S_\xi \xi \leq 1,\, S_\xi > 0\}, \tag{12}$$

$$\delta u \in \Omega_u,\, \Omega_u = \{\delta u \in \mathcal{R}^{n_u} : \delta u^T S_u \delta u \leq 1,\, S_u > 0\}. \tag{13}$$

In addition, n_d external perturbations acting on (9)-(11) can be bounded according to the following ellipsoidal condition

$$d \in \Omega_d,\, \Omega_d = \{d \in \mathcal{R}^{n_d} : d^T M_d d \leq 1,\, M_d > 0\}. \tag{14}$$

Finally, according to [20], it's possible to define the robust *D*-invariant region

$$\Gamma_0 = \{\xi \in \mathcal{R}^{n_\xi} : \xi^T P_0 \xi \leq 1 \;\; P_0 > 0\}. \tag{15}$$

Region (15) is robust to the disturbance d and defines a positively invariant ellipsoidal set in the augmented state space, ensuring $\xi \in \Gamma_0$ for all $t \geq 0$, despite uncertainties and admissible disturbances affecting (9)-(11), and thus guaranteeing the feasibility property to the trajectories.

III. SET-THEORETIC TIME-BASED TRAJECTORY SYNCHRONIZATION METHOD

Collision avoidance and coordination of the skid-steered platforms composing the MRS are addressed in this work by synchronizing the platforms through time scheduling of robot departures, ensuring a minimum safety distance between the robots during the trajectory tracking. The procedure consists of *offline*, and *online* phases.

A. Offline phase

In the *offline* phase, mutual delays between platforms that ensure no collision during robot movement are calculated on the basis of planned feasible trajectories. Assume that for each robotic platform a trajectory is preliminarily planned. The trajectory is composed of a sequence of straight segments that must be traveled by the robot at constant assigned forward velocity v_D. Our approach leverages the feasibility properties of the planned trajectories, calculated using the algorithm proposed in [16], in order that for each trajectory the following feasibility condition holds:

- *if the initial state of the system* $\xi(t_0) \in \Gamma_0$*, then* $\xi(t_k) \in \Gamma_k \subseteq \Gamma_0 \; \forall t_k \geq t_0$*, regardless of the allowable uncertainty of (9)-(11) and the external disturbances* (14)*, along the complete trajectory*, see [16].

The feasible trajectory $\mathcal{T}$ is represented by a succession of the robot's desired pose, hereinafter labeled as waypoints:

$$\mathcal{T} = [w(t_0), \cdots, w(t_k), \cdots, w(t_\epsilon)] \in \mathrm{R}^{(\epsilon+1)\times 3}, \tag{16}$$

each of them with the following form $w(t) = [x(t)\;\; y(t)\;\; \theta(t)]$, and where $w(t_0)$ $(w(t_\epsilon))$ is the starting (ending) desired pose. For each couple of planned trajectories $\mathcal{T}_i$ and $\mathcal{T}_j$ where $i, j = 1, \cdots, N$, and $j \neq i$ robotic platforms, with $\mathcal{T}_i(t_0) \neq \mathcal{T}_j(t_0)$, and $\mathcal{T}_i(t_\epsilon) \neq \mathcal{T}_j(t_\epsilon)$, the following procedure is applied.

a) ***Procedure****:* Consider two generic trajectories

$$\mathcal{T}_i = \{S_i, \cdots, A_i, B_i, \cdots F_i\}, \tag{17}$$

$$\mathcal{T}_j = \{S_j, \cdots, A_j, B_j, \cdots F_j\}, \tag{18}$$

where S_i (S_j) is the starting pose of the i-th (j-th) trajectory, and F_i (F_j) is the final pose of the i-th (j-th) trajectory, with A_i, B_i, $(A_j,\; B_j)$ intermediate waypoints. It is possible to calculate the length of the segment $\overline{AB}$ as follows

$$l_{AB} = \sqrt{(x_A - x_B)^2 + (y_A - y_B)^2}. \tag{19}$$

Let N_{AB} be the maximum positive integer for which the following inequality holds

$$N_{AB} \leq \frac{l_{AB}}{v_D \cdot T_s} \tag{20}$$

where T_s is the sampling time. The trajectory segments composing $\mathcal{T}_i$, are discretized using the interpolation parameter $\alpha = \{0, \frac{1}{N_{AB}}, \frac{2}{N_{AB}}, \cdots, 1\}$, such that

$$w(\alpha) = (1-\alpha)A_i + \alpha B_i. \tag{21}$$

For each pair of segments, the extremes of the interval in which the points belonging to the D-invariant ellipsoids (15) centered on the waypoints of the trajectories, and appropriately oriented, are found to be less than a distance that guarantees no collision. To find these extremes the following optimization problems are solved.

1) Collision lower bound

$$\underline{\beta} = \min_{\alpha} \beta \tag{22}$$

s.t.

$$\beta \geq 0,\; \beta \leq 1, \tag{23}$$

$$\begin{bmatrix} 1 & R(\theta_\alpha)^T (w_i - w(\alpha))^T \\ * & \tilde{P}_0^{-1} \end{bmatrix} \geq 0 \tag{24}$$

$$\begin{bmatrix} 1 & R(\theta_\beta)^T (w_j - w(\beta))^T \\ * & \tilde{P}_0^{-1} \end{bmatrix} \geq 0 \tag{25}$$

$$\begin{bmatrix} 2\gamma & (w_i - w_j)^T M^T \\ * & I \end{bmatrix} \geq 0 \tag{26}$$

2) Collision upper bound

$$\overline{\beta} = \max_{\alpha} \beta \tag{27}$$

s.t.

$$\beta \geq 0,\; \beta \leq 1, \tag{28}$$

$$\begin{bmatrix} 1 & R(\theta_\alpha)^T (w_i - w(\alpha))^T \\ * & \tilde{P}_0^{-1} \end{bmatrix} \geq 0 \tag{29}$$

$$\begin{bmatrix} 1 & R(\theta_\beta)^T (w_j - w(\beta))^T \\ * & \tilde{P}_0^{-1} \end{bmatrix} \geq 0 \tag{30}$$

$$\begin{bmatrix} 2\gamma & (w_i - w_j)^T M^T \\ * & I \end{bmatrix} \geq 0 \tag{31}$$

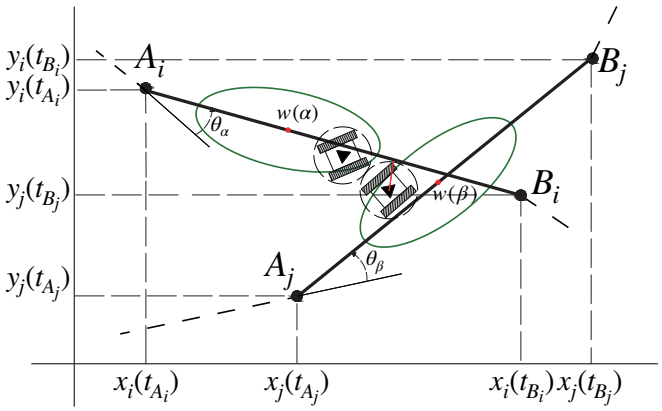

Fig. 2. 2D Graphical representation. $\overline{A_iB_i}$ and $\overline{A_jB_j}$ represent two segments of two different trajectories. The red dots represent $w(\alpha)$ and $w(\beta)$. Green ellipsoids are a 2D representation of the (15) sets to which the state of the system is guaranteed to belong. The solid red line represents γ, the value accounting for the physical dimensions of the robots. The times t_{A_i} (t_{A_j}) and t_{B_j} (t_{B_j}) correspond to the time instants associated with the respective positions.

where

$$w(\beta) = (1-\beta)A_j + \beta B_j, \tag{32}$$

and $\tilde{P}_0$, following [21], is the inner approximation of the projection onto the components x, y, and θ of the ellipsoidal set (15). In addiction, $R(\theta_\alpha)$ and $R(\theta_\beta)$ are rotational matrices defined as in eqs. (7), the matrix M is the following

$$M = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

and γ is the minimum distance accounting for the physical dimensions of the robotic platforms calculated as follows:

$$\gamma = \sqrt{\left(\frac{h_1}{2}\right)^2 + \left(\frac{h_2}{2}\right)^2} \tag{33}$$

with h_1,h_2 being the width and length of the robot. See Fig. 2 for a 2D simplified graphical representation.

*b) **Delays**:* The minimum and maximum values of β resulting from the minimization and maximization problems represent the extremes of the range where the collisions could occur. By considering that the trajectories are traversed by the robots at assigned constant forward velocity v_D, it is possible to calculate the instants of time corresponding to the values of $\underline{\beta}$ and $\overline{\beta}$ as:

$$t(\beta) = \frac{\sqrt{(x_\beta - x_{A_j})^2 + (y_\beta - y_{A_j})^2}}{v_D} + t_{A_j}, \tag{34}$$

where t_{A_j} is the instant of time when the robot reaches A_j. Calculating the global minimum $t(\underline{\beta})$ and maximum $t(\overline{\beta})$, that considers all the segments that make up the trajectories, the minimum $\underline{\Delta}_{i,j}$ and maximum $\overline{\Delta}_{i,j}$ delay value between the departure times of the skid-steered robotic platforms are calculated as follows:

$$\Delta_{i,j} = t_{A_i} + t(\alpha) - t_{A_j} - t(\beta), \tag{35}$$

and finally organized into two matrices $\underline{\mathbf{\Delta}}$ and $\overline{\mathbf{\Delta}}$ with the following form:

$$\mathbf{\Delta} = \begin{bmatrix} 0 & \Delta_{1,2} & \cdots & \Delta_{1,i} & \Delta_{1,j} & \cdots & \Delta_{1,N} \\ \Delta_{2,1} & 0 & \cdots & \Delta_{2,i} & \Delta_{2,j} & \cdots & \Delta_{2,N} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \Delta_{i,1} & \cdots & \cdots & 0 & \Delta_{i,j} & \cdots & \Delta_{i,N} \\ \Delta_{j,1} & \cdots & \cdots & \Delta_{j,i} & 0 & \cdots & \Delta_{j,N} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \Delta_{N,1} & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \end{bmatrix} \tag{36}$$

These values represent the minimum and maximum time that the j-th robot can/should wait before starting up, taking into account the presence of the i-th robot. Specifically, the j-th robot can go at most up to $\underline{\Delta}_{i,j}$ seconds later, or it must wait to leave at least $\overline{\Delta}_{i,j}$ seconds to ensure no collisions. The waiting time determines the order in which the robots pass through the critical sections of the trajectory; it is possible to obtain negative values of $\Delta_{i,j}$, which represent the time advance that one robot must have over the other to avoid collision.

B. Online phase

The delay calculated in the *offline* phase for feasible synchronization of trajectories are used in the *online* phase of the algorithm to schedule the motion starts of the robotic platforms according to the instant of time when the activation request occurs. The delay value is calculated by minimizing the overall duration of the operations. Consider a N robot system. Without loss of generality we assume that $t_{req,N}$ is the instant at which the activation of the N-th robot is requested. Let $t_{act,i}$, with $i = 1, \cdots, N-1$, be the startup times of the first $N-1$ robots. With the aim of freeing up the operational space in the shortest possible time, thus minimizing the total time for carrying out operations, the startup time of the N-th robot is calculated according to the following linear programming problem that guarantees the minimum delay with respect to $t_{req,N}$

$$\Delta_{t_N} = min \Delta_t \tag{37}$$

s.t.

$$\Delta_t \geq 0 \tag{38}$$

$$t_{req,N} + \Delta_t - t_{act,i} \leq \underline{\Delta}_{N,i},\ i = 1, \cdots, N-1 \tag{39}$$

or

$$\Delta_t \geq 0 \tag{40}$$

$$t_{req,N} + \Delta_t - t_{act,i} \geq \overline{\Delta}_{N,i},\ i = 1, \cdots, N-1 \tag{41}$$

IV. Numerical Results

This section presents the results of several numerical simulations. The skid-steered robots are subject to model uncertainties and external disturbances due to varying terrain sliding coefficients, as reported in Sect. II. The platforms are remotely controlled, accounting for delays in the control chain due to the communication network, with a randomly

TABLE I

SIMULATION RELEVANT VALUES.

Parameter	Minimum value	Maximum value	Unit
e_x	-0.35	0.35	m
e_y	-0.45	0.45	m
e_θ	-60	60	deg
V	0	0.4	m/s
ω	-35	35	deg/s
τ	0	$400 \cdot 10^{-3}$	s
t_{act}	0	60	s

varied initial tracking error. The virtual operating environment represents an indoor space with three distinct areas, and is characterized by several slip coefficients whose minimum and maximum values have a deviation of $\pm 25\%$ from the nominal value. The nominal forward velocity assigned to robotic platforms is set at $0.2\ m/s$. All relevant values for figuring out the simulation are given in Tab. I. By considering a physical footprint of each robot of $0.50 \times 0.70\ m^2$, the minimum distance required to avoid collisions results in $2\gamma = 0.86\,m$. Finally, in accordance with Sect. III, the calculation of the delay matrices $\overline{\Delta}$ and $\underline{\Delta}$ gave the following results:

$$\underline{\Delta} = \begin{bmatrix} 0 & -109.15 & -154.34 \\ -101.71 & 0 & -169.42 \\ -56.07 & -92.01 & 0 \end{bmatrix}, \quad \overline{\Delta} = \begin{bmatrix} 0 & 7.88 & -5.01 \\ 54.81 & 0 & 4.32 \\ 110.14 & 83.64 & 0 \end{bmatrix} \tag{42}$$

Each planned trajectory was simulated 10 times, and, in every simulation, the activation request times of robots #2 and #3 were randomly varied with a Gaussian distribution in the range $t_{act} = [0, 60]\ s$, while robot # 1 is always activated in time instant $t_{act} = 0$. The relevant timing data of the simulations are summarized in Tab. II. The results of the simulations are illustrated in Figs. 3-6, which highlight key aspects of the method.

Fig. 3 showcases the inter-platform distances during the simulations. Notably, the collision avoidance safety distance 2γ was always maintained, even in scenarios where the trajectories intersected or the robots operated in close proximity. This result underscores the effectiveness of the synchronization algorithm in coordinating the robots' movements while adhering to safety constraints.

Fig. 4 shows the trajectory tracking errors for all robots during the simulations. These errors remained consistently within the allowable limits, demonstrating the robustness of the feasible planning algorithm used, a key aspect for the application of the proposed solution.

Fig. 5 provides profiles of the forward and rotational velocities of the control algorithm. As can be seen, the values remain in the assigned range, demonstrating the effectiveness of the method even in the case of limited actuation capabilities.

Finally, to emphasize the aspect of coordination, Fig. 6 shows the pose of the robots at the time when the minimum distance between robot #2 and robot #3, corresponding to $t = 190\,s$, of a specific collected Run, was recorded. The results are based on the data collected during Run #8. The ellipsoidal sets (15) to which the robot pose belongs are

TABLE II

RELEVANT SIMULATIONS TIMING VALUES.

	Robot #2		Robot #3	
Run #	Request time [s]	Activation time [s]	Request time [s]	Activation time[s]
1	41.03	41.56	47.02	168.72
2	33.10	33.61	35.01	160.78
3	8.27	10.36	13.06	137.54
4	2.50	10.36	6.41	137.53
5	4.95	10.37	43.17	137.54
6	21.27	21.61	58.27	148.78
7	39.74	40.03	14.64	167.20
8	40.81	41.22	31.67	168.38
9	36.15	36.74	45.03	163.90
10	4.01	10.36	56.36	137.50

highlighted in the figure. It can be seen that even when they turn out to be closest, the minimum distance imposed by the procedure is correctly met. Overall, these results validate the proposed method as a reliable solution for synchronization and coordination of multi-robot trajectories under realistic conditions, confirming the method's ability to ensure safety, accuracy, and robustness in various operational scenarios.

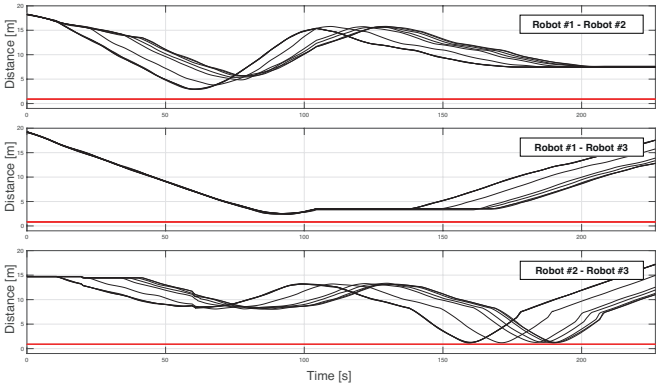

Fig. 3. Actual distances between the robots. The solid red line highlights the threshold value 2γ.

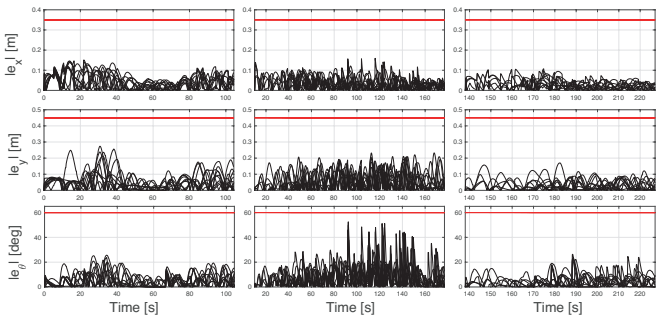

Fig. 4. Trajectory tracking errors. From left to right the values recorded for robot #1, #2, and #3 respectively. The solid red lines represent allowable bounds.

V. CONCLUSION

This paper proposes an approach for the synchronization of the units of a multi-robot systems subjected to constraints, uncertainties and external disturbances. The approach exploits the feasibility properties of the planned trajectories to compute the temporal scheduling of the robots departures.

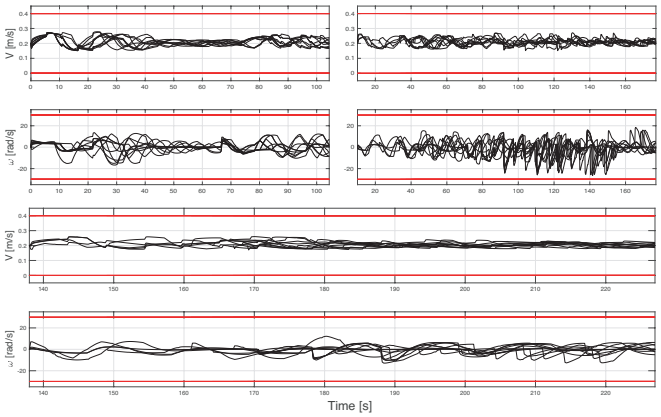

Fig. 5. Values of the forward and rotational velocities of the robots. The plots in the top-left, top-right, and bottom-center show the values for robots # 1, # 2, and # 3, respectively. The solid red lines represent allowable bounds.

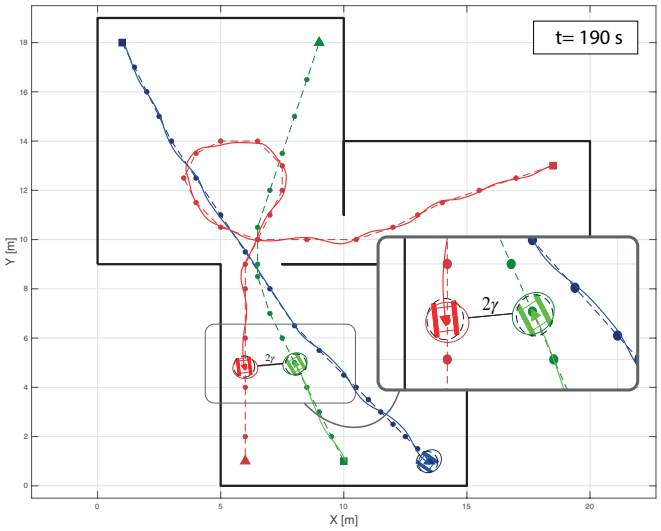

Fig. 6. Run #8. Snapshot at the time instant $t = 190\,s$. Dashed line represents assigned trajectories for the robot #1, #2, and #3. Solid lines represents the executed trajectories. Squares and triangles represent the starting and target points. The blue, red, and green ellipsoids represent the D-invariant regions for robot #1, #2, and #3, respectively. Black dashed circles account for the physical footprints of the robot and have radius γ. The black line between the ellipsoids represents the distance recorded, which turns out to be equal to the threshold for collision avoidance 2γ.

The algorithm considers two phases: an offline phase, where the delay intervals useful for synchronization are calculated on the basis of the planned trajectories, and an online phase, where the actual departure delay is calculated solving a linear optimization problem that minimizes the total time of occupation of the operational space by the platforms. The effectiveness of the proposed method was demonstrated through a series of numerical simulations, involving three robotic platforms subjected to uncertainties and external disturbances in an indoor environment.

REFERENCES

[1] A. A. A. Rasheed, M. N. Abdullah, and A. S. Al-Araji, "A review of multi-agent mobile robot systems applications," *International Journal of Electrical and Computer Engineering*, vol. 12, no. 4, pp. 3517–3529, 2022.

[2] F. Fahimi, H. Ashrafiuon, and C. Nataraj, "Obstacle avoidance for spatial hyper-redundant manipulators using harmonic potential functions and the mode shape technique," *Journal of Robotic Systems*, vol. 20, no. 1, pp. 23–33, 2003.

[3] S. R. Bassolillo, E. D'Amato, S. Iacono, S. Pennino, and A. Scamardella, "Trajectory planning of a mother ship considering seakeeping indices to enhance launch and recovery operations of autonomous drones," in *Oceans*, vol. 5, no. 3. MDPI, 2024, pp. 720–741.

[4] V. Scordamaglia, A. Ferraro, F. Tedesco, and G. Franzè, "Set-theoretic approach for autonomous tracked vehicles involved in post-disaster first relief operations," in *2024 10th International Conference on Control, Decision and Information Technologies (CoDIT)*. IEEE, 2024, pp. 2006–2011.

[5] V. Scordamaglia and A. Ferraro, "A set-based method for planning coordinated trajectories for skid-steered robotic units subject to constraints, uncertainties and external disturbances," *IEEE Access*, 2025.

[6] V. Digani, L. Sabattini, C. Secchi, and C. Fantuzzi, "Towards decentralized coordination of multi robot systems in industrial environments: A hierarchical traffic control strategy," in *2013 IEEE 9th International Conference on Intelligent Computer Communication and Processing (ICCP)*. IEEE, 2013, pp. 209–215.

[7] A. Ferraro, C. De Capua, and V. Scordamaglia, "Integrated trajectory planning and fault detection for a skid-steered mobile robot deployed in post-disaster scenarios," in *2024 IEEE International Workshop on Technologies for Defense and Security (TechDefense)*. IEEE, 2024, pp. 413–418.

[8] S. R. Bassolillo, E. D'Amato, and I. Notaro, "A consensus-driven distributed moving horizon estimation approach for target detection within unmanned aerial vehicle formations in rescue operations," *Drones*, vol. 9, no. 2, p. 127, 2025.

[9] Y. Guo, L. Yu, and J. Xu, "Robust finite-time trajectory tracking control of wheeled mobile robots with parametric uncertainties and disturbances," *Journal of Systems Science and Complexity*, vol. 32, no. 5, pp. 1358–1374, 2019.

[10] A. Ferraro and G. Martino, "Time-delay approach to coordinate movements of mobile robots subject to uncertainties and external disturbances," in *2024 IEEE International Workshop on Technologies for Defense and Security (TechDefense)*. IEEE, 2024, pp. 462–467.

[11] S. R. Bassolillo, G. Raspaolo, L. Blasi, E. D'Amato, and I. Notaro, "Path planning for fixed-wing unmanned aerial vehicles: An integrated approach with theta* and clothoids," *Drones*, vol. 8, no. 2, p. 62, 2024.

[12] J. Peng and S. Akella, "Coordinating multiple robots with kinodynamic constraints along specified paths," *The international journal of robotics research*, vol. 24, no. 4, pp. 295–310, 2005.

[13] M. Kloetzer, C. Mahulea, and J.-M. Colom, "Petri net approach for deadlock prevention in robot planning," in *2013 IEEE 18th Conference on Emerging Technologies & Factory Automation (ETFA)*. IEEE, 2013, pp. 1–4.

[14] S. Liu, J. Shen, W. Tian, J. Lin, P. Li, and B. Li, "Balanced task allocation and collision-free scheduling of multi-robot systems in large spacecraft structure manufacturing," *Robotics and Autonomous Systems*, vol. 159, p. 104289, 2023.

[15] X. Wang, M. Kloetzer, C. Mahulea, and M. Silva, "Collision avoidance of mobile robots by using initial time delays," in *2015 54th IEEE Conference on Decision and Control (CDC)*. IEEE, 2015, pp. 324–329.

[16] A. Ferraro and V. Scordamaglia, "A set-based approach for detecting faults of a remotely controlled robotic vehicle during a trajectory tracking maneuver," *Control Engineering Practice*, vol. 139, p. 105655, 2023.

[17] L. Bakule, "Decentralized control: An overview," *Annual reviews in control*, vol. 32, no. 1, pp. 87–98, 2008.

[18] S. Choi, E. Kim, K. Lee, and S. Oh, "Real-time nonparametric reactive navigation of mobile robots in dynamic environments," *Robotics and Autonomous Systems*, vol. 91, pp. 11–24, 2017.

[19] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear matrix inequalities in system and control theory*. SIAM, 1994.

[20] I. Kolmanovsky, E. G. Gilbert *et al.*, "Theory and computation of disturbance invariant sets for discrete-time linear systems," *Mathematical problems in engineering*, vol. 4, pp. 317–367, 1998.

[21] G. Franzè, F. Tedesco, and D. Famularo, "Model predictive control for constrained networked systems subject to data losses," *Automatica*, vol. 54, pp. 272–278, 2015.