IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 7 March 2025, accepted 20 March 2025, date of publication 27 March 2025, date of current version 7 April 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3555494

RESEARCH ARTICLE

A Set-Based Method for Planning Coordinated Trajectories for Skid-Steered Robotic Units Subject to Constraints, Uncertainties, and External Disturbances

VALERIO SCORDAMAGLIA, (Member, IEEE), AND ALESSIA FERRARO
DIIES Department, Universitá degli Studi Mediterranea di Reggio Calabria, 89122 Reggio Calabria, Italy

Corresponding author: Valerio Scordamaglia (valerio.scordamaglia@unirc.it)

This work was supported by the "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 under Grant P2022XER7W.

ABSTRACT This paper introduces a method for planning coordinated trajectories for a group of remotely-controlled, skid-steered robotic units with limited motion capabilities, subject to skidding and slipping phenomena. The proposed solution aims to define optimal trajectories that can be simultaneously tracked by the robots while ensuring collision-free motion, even in the presence of unavoidable tracking errors due to the limited authority of the control algorithm. A key feature is that, at the cost of certain degree of conservatism, the robotic units do not require collision avoidance capabilities to solve online coordination problems during motion. From a methodological point of view, the problem of planning robust and coordinated trajectories that ensure collision-free robot movements is formulated as a graph search problem involving semidefinite programming with constraints in the form of linear matrix inequalities. Finally, in order to show the effectiveness of the proposed solution and highlight some of its potential limitations, the results of several numerical and experimental simulations are discussed.

INDEX TERMS Multirobot systems, coordinated feasible trajectories planning, set-theoretic methods, uncertain systems, semidefinite programming, linear matrix inequalities.

I. INTRODUCTION

A. BACKGROUND

The ability to use autonomous multi-robot systems (MRS) to perform complex tasks is a key requirement in various application scenarios such as reconnaissance [1], surveillance and patrolling of large areas [2], search and rescue operations after disasters [3]. In these scenarios, it is crucial for effectiveness that several robot units can operate in a joint deployment scenario. This requirement necessitates the planning of coordinated movement strategies that allow each robot unit to reach its goal while avoiding collisions. When planning the movement of an MRS system, not only the obstacles in the environment must be taken into account, but also the possible interference between the robots. When robots in a team perform independent tasks in a common workspace, each of them becomes a moving obstacle for the others. The motion planning of each robot in the team must therefore take into account the movements of the others. Motion planning for the units of an MRS system is inherently difficult, and research in this area has been going on for more than two decades [4]. The most common solution used so far in various applications involves a decoupled approach, where trajectory planning methods [5], [6] are used for motion planning that allow each robot unit to fulfill its assigned task, without taking into account the presence of other units, while the aspect of coordination is handled on-line by implementing reactive capabilities through the controllers so that the robots are able to coordinate with other units in real time avoiding collisions. In the following, a literature overview is proposed.

Motion planning - It generally involves determining a trajectory to reach the final destination while avoiding obstacles

The associate editor coordinating the review of this manuscript and approving it for publication was Mouloud Denai.

and optimizing certain criteria such as distance, time or energy consumption. Classical solutions can be roughly divided into a) graph-based algorithms, b) sampling-based algorithms, c) potential field methods and d) optimization-based approaches.

Graph-based algorithms, such as Dijkstra's algorithm and the A* algorithm, discretize the environment into a grid or graph and search for the shortest path by using heuristic functions to optimize computational efficiency [7]. *Sampling-based algorithms*, including the Probabilistic Roadmap Method (PRM) and Rapidly-exploring Random Trees (RRT), tackle high-dimensional spaces by randomly sampling the environment to create a roadmap or tree representing viable paths [8], [9]. In recent years, these classical methods have been significantly improved to increase their performance and applicability. In [10] an asymptotically optimal version of RRT was proposed, which guarantees convergence to the optimal path when the number of samples approaches infinity. Similarly, PRM* was introduced as an optimal variant of PRM that provides both probabilistic completeness and asymptotic optimality [10]. *Potential field methods*, which model the robot and obstacles as charged particles, are often used to plan motions that take into account the presence of obstacles due to their simplicity [11]. However, these methods often encounter problems with local minima. To solve this problem, hybrid approaches have been explored that combine potential fields with heuristic solutions [12]. *Optimization-based methods*, on the other hand, formulate trajectory planning as a mathematical optimization problem that leads to a continuous path without the computational complexity increasing exponentially as the operation space expands [13].

A common criticism that affects classical approaches is that they are unable to solve motion planning problems explicitly taking into account the limited motion capabilities of robots, and/or external disturbances and constraints that may affect them. For such reasons, recent literature has explored new motion planning methods that rely on set-theoretical principles [14], [15], [16], [17]. This is mainly due to the fact that such method for motion planning make it easy to take into account robot actuation constraints and the presence of external disturbances that may affect the trajectory tracking capabilities. In particular, set-based methods provide a powerful solution to the motion planning problem by leveraging mathematical set representations to handle uncertainty and ensure robustness. There are several applications in the literature that use set-based approaches for motion planning. In [18], motion primitives are used to provide a structured approach for trajectory planning by discretizing the motion possibilities into a predefined set of maneuvers. Set-based motion primitives account for model uncertainties and tracking errors by incorporating reachable sets. This allows for improved robustness in dynamically changing environments. In [14] and [19], the concept of reachability analysis is used. It calculates the set of all states that a system can reach within a given time horizon under bounded inputs and disturbances. This kind of approach is often used in automated driving to define safe driving corridors and efficiently detect collision risks. In [20], set-based methods for intent-expressive motion planning were explored, where an agent's trajectory is designed to be interpretable by other agents or human operators. This approach is particularly important for human-robot cooperation and the coordination of multiple agents. In [21] an approach based on invariant sets is used. Invariant sets define a region of the state space in which the system remains within safe constraints. These sets are used to ensure compliance with constraints when computing trajectories and thus enable safe and robust navigation in constrained environments In [22] the problem of trajectory planning for multi-agent systems is addressed. In multi-agent systems, distributed set-based planning allows each agent to autonomously compute its trajectory while ensuring collision avoidance and cooperative behavior. This approach is particularly useful for autonomous vehicles operating in complex traffic environments. Although they are very effective in dealing with problems of model uncertainty and the presence of disturbances, set-based methods suffer from the following common weaknesses.

- Conservatism: set-based approaches can be very conservative in defining feasible and collision-free motions, which can significantly reduce the efficiency of the trajectory and its ultimate existence [18].
- Computational complexity: Set-based approaches involve high-dimensional set computations, and this aspect often limits the usability for offline applications [18], [19].
- Scalability: this weakness, closely related to computational complexity, limits the effectiveness of set-based approaches in scenarios with a limited number of autonomous vehicles operating under dynamic and uncertain conditions [22], [23].

Motion execution and reactive control strategies- Motion execution strategies include all types of algorithms that are used to execute a desired motion. They typically arise in the development of control laws so that each robot is able to follow a desired path with a given timing law. This problem can become challenging when the robot model is uncertain, actuation capabilities are limited, or external disturbances affect the robot's ability to track the trajectory. In [24] a solution to the problem of trajectory tracking for autonomous, underactuated vehicles in the presence of model parameters is proposed. The paper shows how adaptive supervisory switching control can be combined with a Lyapunov-based nonlinear control law for trajectory tracking to ensure global bounding and convergence of the position tracking error in an arbitrarily small neighborhood of the origin. In [25], the problem is addressed by developing a robust adaptive control scheme for trajectory tracking of underactuated surface vessels (USVs) under unknown dynamics, external disturbances, and input saturation. In [26], a recent survey focusing on motion execution strategies is proposed.

In applications where several robot units have to share the same operating space, the problem of motion planning must be combined with an effective coordination strategy. In the literature, they are generally implemented by decentralized strategies that allow each robot to make decisions based on local information, which improves scalability and reduces computational costs [27]. From the application perspective, [28] has studied the coordination and collective movement of multi-agent robotic systems using flocking models that consider the influence of cooperative and competitive interactions as well as communication delays in ensuring group cohesion and collision-free movement. In [29], the problem of secure synchronous tracking control in networked agent systems under antagonistic interactions and denial-of-service attacks was addressed, where a distributed secure control scheme ensures robust trajectory tracking even under adverse conditions. The reactive control strategies enable robots to quickly react to dynamic environments and unexpected obstacles, and ensure flexible deployment in complex spaces. Among the solutions proposed in the literature, artificial potential fields model robots and obstacles using attractive and repulsive forces that guide the robots to their goals while avoiding collisions [11]. In behavior-based control, robot actions are decomposed into behaviors such as obstacle avoidance and target search, which are then combined to produce the desired motion [30]. Control barrier functions provide a mathematical framework to ensure that safety conditions are met and control goals are achieved [31]. Other reactive control strategies include: the dynamic window approach [32], which takes into account the kinematic constraints of the mobile robot; the elastic band method [33], which adjusts the path of the robot based on the shape of obstacles; and the null-space based behavioral approach [34], where collision avoidance is formulated by coordinated control using a priority logic. Finally, collision avoidance can also be addressed by implementing algorithms that regulate the speed of each robot along its assigned path. The decomposition of the autonomous navigation problem into separate subproblems of path planning and robot velocity profile assignment is discussed e.g. in [35] and [36].

B. MOTIVATIONS AND CONTRIBUTION

Although decoupled independent motion planning approach combined with reactive control systems is effective for collision management in many applications [37], [38], it has its limitations in certain conditions. In particular, when multiple robotic units must operate simultaneously in constrained or cluttered environments, reactive collision avoidance algorithms cannot reliably ensure mission success. Moreover, in time-critical missions, the relying on reactive strategies may be insufficient to guarantee the achievement of the assigned goals [39], [40] and at the same time, the use of decentralized collision avoidance algorithms drastically reduces the autonomy of the robots. To address all of the above problems, in this paper we present a solution to these challenges that focuses on motion planning for a team of skid-steered autonomous robots operating in a shared, confined space. The key assumption is that no reactive control strategies are implemented for collision avoidance. Instead, the proposed trajectory planning algorithm utilizes set-theoretic principles to achieve two main goals: (i) planning motions defined in terms of paths with associated travel time laws that the robot units can follow, ensuring a predefined maximum tracking error is explicitly guaranteed by taking into account the motion capabilities of the robots; (ii) planning coordinated motions that allow the robots to move without collisions.

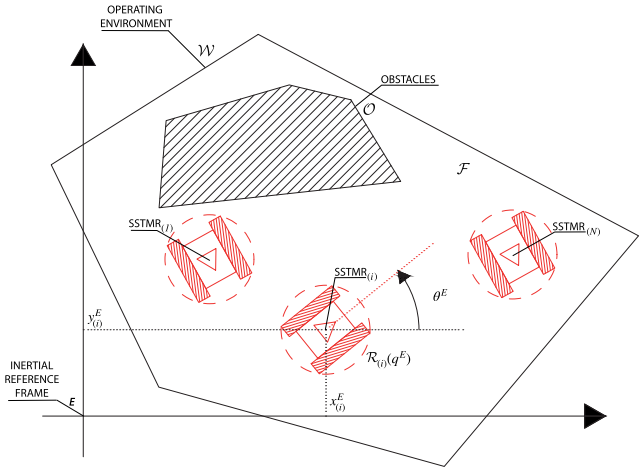

FIGURE 1. The inertial reference system is denoted by E, $\mathcal{W}$ represents the operating environment, $\mathcal{O}$ denotes the occupied space, $\mathcal{F}$ denotes the free space, $\mathbf{q}^E$ denotes the pose of the generic SSTMR, $\mathcal{R}(\mathbf{q}^E)$ denotes the region occupied by SSTMR whose pose is $\mathbf{q}^E$.

II. PROBLEM STATEMENT

Consider the multi-robot system consisting of N remote-controlled, autonomous, skid-steered tracked units, denoted as $\{\text{SSTMR}_{(1)}, \text{SSTMR}_{(2)}, \cdots, \text{SSTMR}_{(N)}\}$ operating in a common space $\mathcal{W} \subseteq \mathbf{R}^2$. Let $\mathcal{W}$ be connected. Suppose that $\mathcal{W}$ is also known. Consider an inertial reference frame $\mathbf{E}$. Let $\mathbf{q}^E = [x^E\ y^E\ \theta^E]^T$ be a pose vector expressed in $\mathbf{E}$, defined by the position $\{x^E, y^E\}$ and the direction θ^E, see Fig. 1.

Let us denote with $\mathcal{A}(\mathbf{q}^E) = \{x^E, y^E\}$ and with $\mathcal{Y}(\mathbf{q}^E) = \theta^E$. Let $\mathcal{O}$ denote the space occupied by obstacles in $\mathcal{W}$. Suppose that $\mathcal{O}$ is time invariant. Let $\mathcal{F} = \mathcal{W} \ominus \mathcal{O}$ be the free space available to the robots. To define the space occupied by the robot body, we consider a circle of radius $\mathcal{R}$. The space occupied by the robot with the pose $\mathbf{q}^E$ is denoted by $\mathcal{R}(\mathbf{q}^E)$. Let us define the path $\mathcal{P}^E$ as a sequence of N_P poses

$$\mathcal{P}^E = \{\mathbf{q}^E_{<i>}\}_0^{N_P-1} \tag{1}$$

with $i = 0, \cdots, N_P - 1$. Let $\mathcal{T}_{\mathcal{P}^E}$ be the trajectory obtained by associating a time parametrization with the path $\mathcal{P}^E$. Denote with $\mathcal{T}_{\mathcal{P}^E}(t)$ the pose of the trajectory $\mathcal{T}_{\mathcal{P}^E}$ at time t with $t \in [0, t_{end}]$ where t_{end} is the time required to traverse $\mathcal{T}_{\mathcal{P}^E}$.

We assume that each SSTMR$_{(i)}$ is controlled by the commands of the forward velocity $V_{(i)}$ and the rotational velocity $\omega_{(i)}$. Let

$$\mathbf{u}_{(i)} = [V_{(i)}\, \omega_{(i)}]^T \tag{2}$$

be the vector of control variables.

It is assumed that the robot SSTMR$_{(i)}$ has limited movement possibilities according to the following condition

$$\mathbf{u}_{(i)} \in \mathcal{U}_{(i)} \tag{3}$$

where

$$\mathcal{U}_{(i)} := \Big\{ \{V_{(i)}, \omega_{(i)}\} : \underline{V}_{(i)} \leqslant V_{(i)} \leqslant \overline{V}_{(i)}, \quad \underline{\omega}_{(i)} \leqslant \omega_{(i)} \leqslant \overline{\omega}_{(i)} \Big\}. \tag{4}$$

being $\underline{V}_{(i)}(\underline{\omega}_{(i)})$ and $\overline{V}_{(i)}(\overline{\omega}_{(i)})$ the minimum and maximum values of the forward (rotational) speed, respectively. It is also assumed that the interaction between the tracks and the ground exposes each SSTMR$_{(i)}$ to non-negligible sliding phenomena, so that

$$\hat{\mathbf{u}}_{(i)}(t) \neq \mathbf{u}_{(i)}(t) \tag{5}$$

with $\hat{\mathbf{u}}_{(i)}(t) = \left[\hat{V}_{(i)}(t)\ \hat{\omega}_{(i)}(t)\right]^T$ vector of actual velocities $\hat{V}_{(i)}(t)$ and $\hat{\omega}_{(i)}(t)$ of SSTMR$_{(i)}$ at time t. We also assume that each SSTMR$_{(i)}$ uses its own $\mathcal{C}_{(i)}$ trajectory tracking controller located on a ground-station and connected via a data communication network. The use of remote control contributes significantly to increasing the operational autonomy of the individual robot unit by shifting the entire computational load to a remote ground station. However, it must be pointed out that such a control architecture has significant drawbacks, mainly related to the introduction of latency, which, if not properly taken into account, can affect the operation and performance of the entire system. In this paper, we assume that the time delay caused by the data communication channel can be abstractly summarized in a single positive scalar $\tau_{(i)}$ for each robot

$$\tau_{(i)} \in [0,\ \tau_{max}],\ \ i = 1, \cdots, N \tag{6}$$

representing the total delay in the control loop for the i-th robot. The presence of a latency (6) in the control loop in combination with the limited handling capabilities (3) of the robotic units on the one hand and the sliding phenomena (5) due to the interaction between tracks and ground on the other hand can lead to non-zero trajectory tracking errors. It is expressed by the following relationship

$$\mathcal{E}_{(i)}(t) = \mathbf{q}^E_{(i)}(t) - \mathcal{T}_{\mathcal{P}^E_{(i)}}(t) \tag{7}$$

that represents difference between the pose of SSTMR$_{(i)}$ and the prescribed one at time t. In this paper, we assume that the closed-loop dynamics of the trajectory tracking error, i.e., considering the effect of the controller $\mathcal{C}_{(i)}$, can be described by the following discrete-time uncertain system with norm-bounded uncertainty

$$\xi_{(i)}(t_{k+1}) = \Phi\xi_{(i)}(t_k) + G_\mu\delta\mu_{(i)}(t_k) + G_p\mathbf{p}_{(i)}(t_k) \tag{8}$$

$$\mathbf{p}_{(i)}(t_k) = \Delta_{(i)}(t_k)\sigma_{(i)}(t_k) \tag{9}$$

$$\sigma_{(i)}(t_k) = C_\sigma\xi_{(i)}(t_k) \tag{10}$$

$$\mathcal{E}_{(i)}(t_k) = F\xi_{(i)}(t_k) \tag{11}$$

where $\|\Delta_{(i)}(t_k)\| < 1\ \forall t_k \geqslant 0$ with ϕ, G_μ, C_σ and F matrices of appropriate size, and $\delta\mu$ an external disturbance assumed bounded in accordance with the following ellipsoidal constraint

$$\delta\mu_{(i)} \in \Omega_\mu,\ \Omega_\mu = \{\delta\mu_{(i)} \in \mathcal{R}^{n_\mu} : \delta\mu^T_{(i)}M_\mu\delta\mu_{(i)} \leqslant 1,\ M_\mu > 0\}. \tag{12}$$

The aim of this paper is to find a solution to the following problem.

Coordinated planning of collision-free trajectories for N robots forming the MRS and operating in a shared space $\mathcal{F}$ (**TPF-MRS**)

The goal is to plan a set of trajectories $\{\mathcal{T}_{\mathcal{P}^E_{(1)}}, \mathcal{T}_{\mathcal{P}^E_{(2)}}, \cdots, \mathcal{T}_{\mathcal{P}^E_{(N)}}\}$ that allow the SSTMRs to move at the nominal speed V_D guaranteeing:

1) limited trajectory tracking errors

$$\mathcal{A}\left(\mathbf{q}^E_{(i)}(t)\right) \in \mathcal{Z}\left(\mathcal{T}_{\mathcal{P}^E_{(i)}}(t)\right) \tag{13}$$

$\forall t \geqslant 0$ with $i = 1, \cdots, N\ \forall t \geqslant 0$ denoting by $\mathcal{Z}\left(\mathcal{T}_{\mathcal{P}^E_{(i)}}(t)\right)$ a suitable convex set centered in $\mathcal{T}_{\mathcal{P}^E_{(i)}}(t)$;

2) obstacle avoidance

$$\mathcal{R}\left(\mathbf{q}^E_{(i)}(t)\right) \subseteq \mathcal{F} \tag{14}$$

$\forall t \geqslant 0$ with $i = 1, \cdots, N$;

3) collision avoidance

$$\mathcal{R}\left(\mathbf{q}^E_{(i)}(t)\right) \bigcap \mathcal{R}\left(\mathbf{q}^E_{(j)}(t)\right) = \varnothing \tag{15}$$

$\forall t \geqslant 0$, where $i, j = 1, \cdots, N$ and $j \neq i$.

In order for the **TPF-MRS** problem to be well-defined, we assume that the following conditions are met:

(i) there is a connected shared free space in which robots can operate

$$\mathcal{F} \neq \varnothing \tag{16}$$

(ii) start and end poses ensure that no collisions occur

$$\mathcal{R}\left(\mathcal{T}_{\mathcal{P}^E_{(i)}}(0)\right) \cap \mathcal{R}\left(\mathcal{T}_{\mathcal{P}^E_{(j)}}(0)\right) = \varnothing \tag{17}$$

$$\mathcal{R}\left(\mathcal{T}_{\mathcal{P}^E_{(i)}}(t_{end})\right) \cap \mathcal{R}\left(\mathcal{T}_{\mathcal{P}^E_{(j)}}(t_{end})\right) = \varnothing \tag{18}$$

with $i, j = 1, \cdots, N$ and $j \neq i$

FIGURE 2. Experimental setup involving two jaguar V4 by Dr.Robot.

III. PROPOSED SOLUTION: OVERVIEW

To solve the **TPF-MRS** problem, the following issues are formally addressed: (a) define a suitable mathematical model as in (8)-(11) that describes the dynamics of the trajectory tracking error of each robotic unit by taking into account both the effects of the time delay caused by the communication channel and the time-varying sliding phenomena of the tracks on the ground; (b) computing sufficient conditions that ensure that a trajectory can always be followed by a robot with an assigned dynamics that guarantees (13), despite the possible presence of uncertainties, actuation constraints and external disturbances that alter the robot's motion capabilities; (c) formalization of an algorithm for the planning of a set of coordinated trajectories $\{\mathcal{T}_{\mathcal{P}^E_{(1)}}, \mathcal{T}_{\mathcal{P}^E_{(2)}}, \cdots, \mathcal{T}_{\mathcal{P}^E_{(N)}}\}$, which ensure robust collision-free movements (14)-(15).

The paper is therefore structured as follows. In Sect. IV, the method for embedding the dynamics of the trajectory tracking error $\mathcal{T}_{\mathcal{P}^E}$ for a generic SSTMR, subject to skidding and slipping phenomena and remotely controlled via a data communication network, described by a discrete-time uncertain linear system with norm-bounded uncertainty subject to external disturbances is presented. In Sect. V, some set-theoretic arguments that formally define the concept of feasible trajectory by exploiting the properties of embedding error dynamics of trajectory tracking are given. In the Sect. VI, the algorithm for the coordinated planning of N feasible trajectories compatible with (14) and (15) of minimum total length is presented. Sect. VII illustrates the effectiveness of the proposed approach through some experiments performed with an experimental setup available at the Laboratory of Control Systems of the Mediterranean University of Reggio Calabria, involving two tracked units mod.Jaguar V4 by Dr.Robot, see Fig. 2. Finally, VIII discusses the result of a computational cost analysis carried out on the basis of numerical tests.

IV. NORM-BOUNDED LDI OF THE TRAJECTORY TRACKING ERROR DYNAMICS OF A SKID-STEERED TRACKED MOBILE ROBOT

A. ROBOT KINEMATICS AND TRACKS/SOIL INTERACTION

Consider a generic SSTMR. We assume that the robot moves like a rigid body on a horizontal plane, see Fig. 3. At generic time t, the classical kinematic model applies to describe the motion of the SSTMR in $\mathbf{E}$, where θ^E is the direction and x^E and y^E are the geometric center position of the robot. Let $\hat{V}$ and $\hat{\omega}$ be the actual forward and angular velocities of SSTMR. Thus, we can express the first-order kinematic model of the robot as follows

$$\begin{aligned} \dot{x}^E(t) &= cos\left(\theta^E(t)\right)\hat{V}(t) \\ \dot{y}^E(t) &= sin\left(\theta^E(t)\right)\hat{V}(t) \\ \dot{\theta}^E(t) &= \hat{\omega}(t) \end{aligned} \tag{19}$$

Let V and ω be the commands by which the SSTMR is controlled (2). Due to the sliding of the tracks on the ground, it can happen according to the hypothesis (5) that $V(t) \neq \hat{V}(t)$ and/or $\omega(t) \neq \hat{\omega}(t)$. In order to rewrite (19) as a function of the control velocity vector (2), the interaction between the tracks and the ground must be taken into account. In this paper, the approach firstly proposed in [15] is used to represent this interaction. Let

$$\rho = \left[\rho^r \;\; \rho^l\right]^T \tag{20}$$

be the vector of the angular velocities of the motors that move the left and right tracks of the robot. Let us now consider two bounded, time-varying positive coefficients, which we will refer to below as sliding coefficients and denote by μ^r and μ^l for the right and left tracks respectively. The following condition applies

$$\mu \in \mathcal{H} \tag{21}$$

where

$$\mathcal{H} = \left\{\{\mu^r, \mu^l\} : \underline{\mu}^r \leqslant \mu^r \leqslant \overline{\mu}^r,\; \underline{\mu}^l \leqslant \mu^l \leqslant \overline{\mu}^l\right\}. \tag{22}$$

where $\underline{\mu}^r(\underline{\mu}^l)$ and $\overline{\mu}^r(\overline{\mu}^l)$ are the minimum and maximum values of the right (left) sliding coefficient. We assume that

$$\underline{\mu}^r \leqslant 1 \leqslant \overline{\mu}^r \tag{23}$$

$$\underline{\mu}^l \leqslant 1 \leqslant \overline{\mu}^l \tag{24}$$

The following relationship applies

$$\hat{\mathbf{u}}(t) = JH(t)\rho(t). \tag{25}$$

where

$$H(t) = \begin{bmatrix} \mu^r(t) & 0 \\ 0 & \mu^l(t) \end{bmatrix}, \; J = \begin{bmatrix} \frac{R}{2} & \frac{R}{2} \\ \frac{R}{D} & -\frac{R}{D} \end{bmatrix}$$

where R is the radius of the pinion that connects the electric motor to the track, D is the distance between the two tracks. Under nominal conditions, when the sliding coefficients both experience the unit value $\mu^r = \mu^l = 1$, the motor speeds $\rho(t)$ required to achieve the control velocities $\mathbf{u}(t)$ can be calculated according to the following equation

$$\rho(t) = J^{-1}\mathbf{u}(t). \tag{26}$$

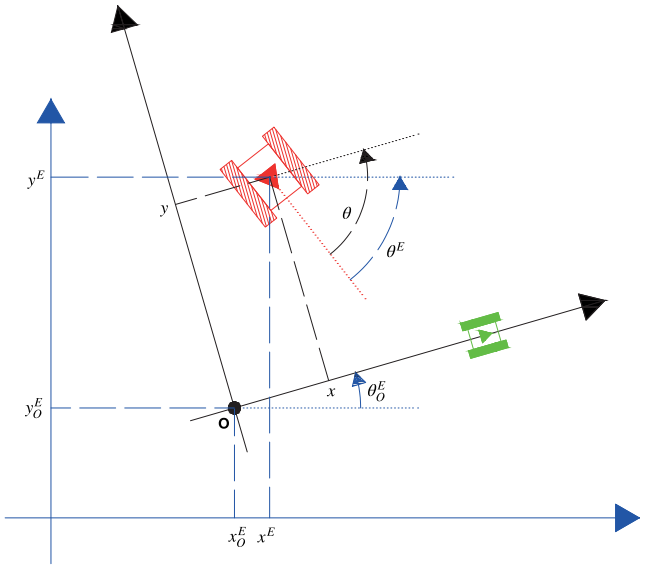

FIGURE 3. The E reference frame is shown in blue, the L reference frame is shown in black. The L reference system has its origin in O. The green robot represents the desired pose of the robot at time t. The red robot represents the actual pose of the robot at time t.The pose of the robot in E is referred to as $\{x^E, y^E, \theta^E\}$. The pose of the robot in L frame is denoted as $\{x, y, \theta\}$.

By recombining the equations (25)-(26), the equation (19) can be rewritten in the following form

$$\dot{\mathbf{q}}^E(t) = G(t)JH(t)J^{-1}\mathbf{u}(t) \tag{27}$$

with $\mathbf{q}^E(t) = [x^E(t)\, y^E(t)\, \theta^E(t)]$ the pose of the SSTMR at time t.

Note that the mathematical model in eq. (27) takes into account the fact that the values of the sliding coefficients $\mu^r(t)$ and $\mu^l(t)$ depend on the interaction between the tracks and the ground and can therefore deviate from the nominal unit condition over time.

B. OPEN-LOOP TRAJECTORY TRACKING ERROR DYNAMICS

To define the dynamics of the trajectory tracking error, consider a given trajectory $\mathcal{T}_{\mathcal{P}^E}$ consisting of a straight line, whose origin is at $O = \mathcal{A}(\mathcal{T}_{\mathcal{P}^E}(0))$ and the direction $\theta_O^E = \mathcal{Y}(\mathcal{T}_{\mathcal{P}^E}(0))$. $\mathcal{T}_{\mathcal{P}^E}$ must be traversed with an assigned forward velocity V_D, see Fig. 3. Let us define a new local reference frame $\mathbf{L}$. It denotes a reference frame centered on O that is rotated with respect to $\mathbf{E}$ by an angle θ_O^E. The following rototranslation applies

$$\mathbf{q}^L(t) = R_E^L\left(\mathbf{q}^E(t) - \mathcal{T}_{\mathcal{P}^E}(0)\right) \tag{28}$$

where

$$R_E^L = \begin{bmatrix} cos\left(\theta_O^E\right) & sin\left(\theta_O^E\right) & 0 \\ -sin\left(\theta_O^E\right) & cos\left(\theta_O^E\right) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{29}$$

From (28) it follows that $\mathcal{T}_{\mathcal{P}^E}(t)$ can be expressed in $\mathbf{L}$ as

$$\mathcal{T}_{\mathcal{P}^L}(t) = \begin{bmatrix} V_D t & 0 & 0 \end{bmatrix}^T \tag{30}$$

with $t \geqslant 0$. Finally, let

$$\mathcal{E}(t) = \mathbf{q}^L(t) - \mathcal{T}_{\mathcal{P}^L}(t) = \begin{bmatrix} \mathcal{E}_x(t) & \mathcal{E}_y(t) & \mathcal{E}_\theta(t) \end{bmatrix}^T \tag{31}$$

be the trajectory tracking error at time t, expressed in $\mathbf{L}$. By applying the classical linearization arguments, it is possible to define the following linear time-invariant mathematical representation

$$\dot{\mathcal{E}}(t) = A\mathcal{E}(t) + B_u\delta_u(t) + B_\mu\delta\mu(t) \tag{32}$$

with

$$A = \left.\frac{\partial\dot{\mathcal{E}}(t)}{\partial q^L(t)}\right|_{\nabla(t)} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & V_D \\ 0 & 0 & 0 \end{bmatrix}, \tag{33}$$

$$B_u = \left.\frac{\partial\dot{\mathcal{E}}(t)}{\partial u(t)}\right|_{\nabla(t)} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}, \tag{34}$$

$$B_\mu = \left.\frac{\partial\dot{\mathcal{E}}(t)}{\partial \mu(t)}\right|_{\nabla(t)} = \begin{bmatrix} \frac{V_D}{2} & \frac{V_D}{2} \\ 0 & 0 \\ \frac{V_D}{D} & -\frac{V_D}{D} \end{bmatrix}, \tag{35}$$

where

$$\nabla(t) = \{\mathcal{T}_{\mathcal{P}^L}(t),\ \mathbf{u}_D(t)\} \tag{36}$$

$\delta_u = \begin{bmatrix} V - V_D & \omega - 0 \end{bmatrix}^T$ where $\mathbf{u}_D(t) = \begin{bmatrix} V_D & 0 \end{bmatrix}^T$ and $\delta\mu = \begin{bmatrix} \mu^r - 1 & \mu^l - 1 \end{bmatrix}^T$. Note that the system (32) describes the linearized trajectory tracking error around the given trajectory (30). All mathematical details can be found in [15].

C. NETWORKED CONTROL SYSTEM

In this paper, we assume that a data communication network is used to implement a trajectory tracking controller. In particular, we assume that the pose information of SSTMR is acquired synchronously at time instants $t_k = kT_S$, where T_S is the sampling time, while the controller and actuators are event-driven and respond to incoming data changes. Based on classical arguments from networked control systems (NCS) theory, the presence of the data communication channel in the control loop was modeled with an uncertain representation that depends on the characteristics of the network. We assume that the number of lost packets during communication can be considered as zero (e.g., a connection-oriented TCP/IP communication protocol is assumed). In this work, all the delays within the control loop are summarized in a single delay term coefficient τ, which represents the end-to-end delay from the sensors to the actuators.

Let τ_{max} be the upper limit of τ. Let $\bar{d}$ be the smallest positive integer that satisfies the relation $\bar{d} \geqslant \tau_{max}/T_s$. According to [41], the following LPV representation can be used to mathematically describe the dynamics of the trajectory tracking error, explicitly taking into account the maximum possible delays caused by the data transmission channel

$$\tilde{\xi}(t_{k+1}) = \overline{A}(\Theta)\tilde{\xi}(t_k) + \overline{B}_u(\Theta)\delta_u(t_k) + \overline{B}_\mu(\Theta)\delta\mu(t_k), \tag{37}$$

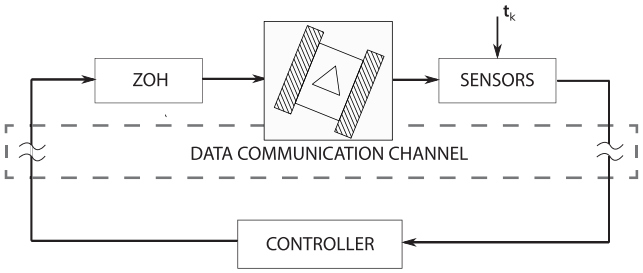

FIGURE 4. An overview of the NCS scheme of a single robot.

where Θ is a suitable parameters vector that depends on the delay, and where

$$\tilde{\xi}(t_k) = \left[\mathcal{E}^T(t_k)\ \delta_u^T(t_{k-1}) \cdots \delta_u^T(t_{k-\bar{d}})\right]^T \in \mathcal{R}^{n_s} \tag{38}$$

is the vector of lifted states at time t_k, hereinafter referred to as *lifted trajectory tracking error*, with $n_s = n_{\mathcal{E}} + \bar{d}n_u$, where $n_{\mathcal{E}}$ and n_u are the number of states and control inputs of (32), respectively. Finally, in accordance with [42], it is possible to resort to the following uncertain linear system with norm-bounded uncertainty, which embeds the dynamics of the LPV system (37)

$$\tilde{\xi}(t_{k+1}) = \tilde{A}\tilde{\xi}(t_k) + \tilde{B}_u\delta_u(t_k) + \tilde{B}_\mu\delta\mu(t_k) + \tilde{B}_p\mathbf{p}(t_k) \tag{39}$$
$$\mathbf{p}(t_k) = \Delta(t_k)\sigma(t_k) \tag{40}$$
$$\sigma(t_k) = \tilde{C}_\sigma\tilde{\xi}(t_k) + \tilde{D}_{\sigma,u}\delta_u(t_k) \tag{41}$$
$$\mathcal{E}(t_k) = \tilde{F}\tilde{\xi}(t_k) \tag{42}$$

where $\|\Delta(t_k)\| < 1\ \forall t_k \geqslant 0$, and $\tilde{B}_p$, $\tilde{C}_\sigma$, $\tilde{D}_{\sigma,u}$ and $\tilde{F}$ are matrices of proper dimensions.

D. CLOSED-LOOP TRAJECTORY TRACKING ERROR DYNAMICS

In order to define the closed-loop trajectory tracking error dynamics, the control law for trajectory tracking must be taken into account. Without loss of generality, we assume that we use a discrete-time LTI controller with the following structure

$$\mathbf{z}(t_{k+1}) = A_C\mathbf{z}(t_k) + B_C\tilde{\xi}(t_k) \tag{43}$$
$$\delta_u(t_k) = C_C\mathbf{z}(t_k) + D_C\tilde{\xi}(t_k) \tag{44}$$

where $\mathbf{z} \in \mathbb{R}^{n_z}$ is the vector of state variables of the control system. Let

$$\xi(t_k) = \left[\tilde{\xi}^T(t_k)\ \mathbf{z}^T(t_k)\right]^T \in \mathbb{R}^{n_\xi} \tag{45}$$

be the augmented vector of the state variables of the closed control loop with $n_\xi = n_z + n_s$. By combining the equations (43)-(44) and (39)-(41), we obtain the following uncertain mathematical model with norm-bounded uncertainty in the presence of external disturbances

$$\xi(t_{k+1}) = \Phi\xi(t_k) + G_\mu\delta\mu(t_k) + G_p\mathbf{p}(t_k) \tag{46}$$
$$\mathbf{p}(t_k) = \Delta(t_k)\sigma(t_k) \tag{47}$$
$$\sigma(t_k) = C_\sigma\xi(t_k) \tag{48}$$
$$\mathcal{E}(t_k) = F\xi(t_k) \tag{49}$$

where $\|\Delta(t_k)\| < 1\ \forall t_k \geqslant 0$, $\phi = \begin{bmatrix} \tilde{A} + \tilde{B}_u D_C & \tilde{B}_u C_C \\ B_C & A_C \end{bmatrix}$,

$$G_\mu = \begin{bmatrix} \tilde{B}_\mu \\ 0_{n_c \times n_\mu} \end{bmatrix}, G_p = \begin{bmatrix} \tilde{B}_p \\ 0_{n_c \times n_s} \end{bmatrix},$$
$$C_\sigma = \left[\tilde{C}_\sigma + \tilde{D}_{\sigma,u}D_C\ \tilde{D}_{\sigma,u}C_C\right],$$

and $F = \left[\tilde{F}\ 0_{n_{\mathcal{E}} \times (\bar{d}n_u + n_c)}\right]$. Furthermore, at the price cost of a certain degree of conservatism, suppose that the constraints (3) and (21) can be recast into the the following ellipsoidal form

$$\xi \in \Omega_\xi,\ \Omega_\xi = \{\xi \in \mathcal{R}^{n_\xi} : \xi^T S_\xi \xi \leqslant 1,\ S_\xi > 0\}, \tag{50}$$
$$\delta\mu \in \Omega_\mu,\ \Omega_\mu = \{\delta\mu \in \mathcal{R}^{n_\mu} : \delta\mu^T M_\mu \delta\mu \leqslant 1,\ M_\mu > 0\}. \tag{51}$$

Finally, consider a given convex constraint on the allowable trajectory tracking error

$$[\mathcal{E}_x\ \mathcal{E}_y]^T \in \mathcal{Z}(0). \tag{52}$$

According to (49), (45), (38) and (31), eq. (52) can be rewritten in the following form

$$\xi \in \Omega_{\mathcal{Z}},\ \Omega_{\mathcal{Z}} = \{\xi \in \mathcal{R}^{n_\xi} : \xi^T S_{\mathcal{Z}} \xi \leqslant 1,\ S_{\mathcal{Z}} \geqslant 0\}. \tag{53}$$

According to (13) this means that

$$\mathcal{A}\left(\mathbf{q}^E(t_k)\right) \in \mathcal{Z}\left(\mathcal{T}_{\mathcal{P}^E}(t_k)\right),\ \forall t_k \geqslant t_0, \tag{54}$$

where $\mathcal{Z}\left(\mathcal{T}_{\mathcal{P}^E}(t_k)\right)$ is the region $\mathcal{Z}(0)$, which is centered in $\mathcal{A}\left(\mathcal{T}_{\mathcal{P}^E}(t_k)\right)$ and rotated by $\mathcal{Y}\left(\mathcal{T}_{\mathcal{P}^E}(t_k)\right)$. Finally, according to [43], we assume that a robust D-invariant ellipsoidal region can be defined

$$\Gamma_0 = \{\xi \in \mathcal{R}^{n_\xi} : \xi^T P_0 \xi \leqslant 1\ \ P_0 > 0\} \subseteq \Omega_\xi \bigcap \Omega_{\mathcal{Z}}, \tag{55}$$

which fulfills the following property: if $\xi(t_0) \in \Gamma_0$ then $\xi(t_k) \in \Gamma_0\ \forall t_k \geqslant t_0$ regardless of the allowable uncertainty of (46)-(48) and the external disturbances (51).

V. TRAJECTORY FEASIBILITY

In this section, we discuss the set-theoretic arguments required to define the sufficient conditions for planning a feasible trajectory for a single SSTMR. For more details, the reader can also read [15]. Consider the path shown in Fig. 5, which consists of a sequence of straight line segments $\overline{W_0W_1}$ and $\overline{W_1W_2}$. Let $\mathbf{E}$ be the reference system centered in W_0 and such that W_1 lies on the x-axis.

Let $\mathbf{E}$ be defined by the following two pose sequences $\mathcal{P}_1^E = \{\mathbf{q}_k^E\}_0^1$ and $\mathcal{P}_2^E = \{\mathbf{q}_k^E\}_2^3$ where $\mathcal{A}(\mathbf{q}_{<0>}^E) = W_0$, $\mathcal{Y}(\mathbf{q}_{<0>}^E) = 0$, $\mathcal{A}(\mathbf{q}_{<1>}^E) = W_1$, $\mathcal{Y}(\mathbf{q}_{<1>}^E) = 0$,

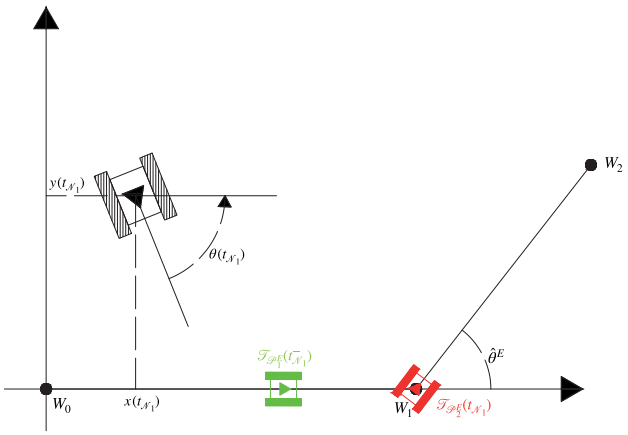

FIGURE 5. Path segment switch from $\overline{W_0W_1}$ to $\overline{W_1W_2}$. The pose of the robot $\mathbf{q}^E(t_{\mathcal{N}_1})$ is shown in black. The green robot represents the pose at time $\mathcal{T}_{\mathcal{P}_1^E}(t_{\mathcal{N}_1}^-)$. The red robot represents the pose $\mathcal{T}_{\mathcal{P}_2^E}(t_{\mathcal{N}_1})$.

$\mathcal{A}(\mathbf{q}^E_{<2>}) = W_1$, $\mathcal{Y}(\mathbf{q}^E_{<2>}) = \hat{\theta}^E$, $\mathcal{A}(\mathbf{q}^E_{<3>}) = W_2$ and $\mathcal{Y}(\mathbf{q}^E_{<3>}) = \hat{\theta}^E$. Let $\mathcal{T}_{\mathcal{P}_1^E}$ and $\mathcal{T}_{\mathcal{P}_2^E}$ be the trajectories obtained from $\mathcal{P}_1^E$ and $\mathcal{P}_2^E$, respectively, using the following time parametrization.

Consider $\mathcal{N}_1$ as the maximum positive integer so that the following inequality holds

$$\mathcal{N}_1 \leqslant \frac{\|W_0 - W_1\|_2}{V_D T_S}. \tag{56}$$

Consider a second reference frame L centered in W_1 and obtained from E by the rototranslation (28) where the matrix R_E^L is defined as in (29) being $\theta_O^E = \hat{\theta}^E$. Let $\mathcal{N}_2$ be the maximum positive integer such that

$$\mathcal{N}_2 \leqslant \frac{\|W_2 - W_1\|_2}{V_D T_S}. \tag{57}$$

Let $\mathcal{T}_{\mathcal{P}_1^E}$ be the trajectory expressed in E for which

$$\mathcal{T}_{\mathcal{P}_1^E}(t) = \begin{bmatrix} V_D t & 0 & 0 \end{bmatrix}^T \tag{58}$$

with $t \in [0, \mathcal{N}_1 T_S)$. The trajectory $\mathcal{T}_{\mathcal{P}_2^E}$ can be expressed more conveniently in the reference frame $\mathbf{L}$ as

$$\mathcal{T}_{\mathcal{P}_2^L}(t) = \begin{bmatrix} V_D(t - \mathcal{N}_1 T_S) & 0 & 0 \end{bmatrix}^T \tag{59}$$

with $t \in [\mathcal{N}_1 T_S, (\mathcal{N}_1 + \mathcal{N}_2) T_S)$.

Definition. The succession of trajectories $\mathcal{T}_{\mathcal{P}_1^E}$ and $\mathcal{T}_{\mathcal{P}_2^E}$ is considered feasible if

$$\xi(t_k) \in \Gamma_0, \ \forall t_k \in [0, (\mathcal{N}_1 + \mathcal{N}_2) T_S]. \tag{60}$$

From a practical point of view, eq. (60) means that the succession of trajectories $\mathcal{T}_{\mathcal{P}_1^E}$ and $\mathcal{T}_{\mathcal{P}_2^E}$ can be tracked with a bounded error, regardless of the presence of uncertainties and disturbances on (46)-(49). To check the condition (60), let

$$S_{W_0} = \{\xi \in \mathcal{R}^{n_\xi} : \xi^T P_{W_0} \xi \leqslant 1 \ \ P_{W_0} > 0\} \subseteq \Gamma_0 \tag{61}$$

be the ellipsoidal set such that $\xi(0) \in S_{W_0}$. Due to the robust positively D-invariance of Γ_0 it follows that $\xi(t_k) \in \Gamma_0 \ \forall t_k \in [0, \mathcal{N}_1 T_S)$. Let us now consider $t_{\mathcal{N}_1} = \mathcal{N}_1 T_S$. This is the time of the switch between the two trajectories $\mathcal{T}_{\mathcal{P}_1^E}$ and $\mathcal{T}_{\mathcal{P}_2^E}$. To guarantee the condition (60), it would suffice to prove that

$$\xi(t_{\mathcal{N}_1}) \in \Gamma_0 \tag{62}$$

because due to the property of robust positively D-invariance of Γ_0 this would imply that $\xi(t_k) \in \Gamma_0 \ \forall t_k \in [\mathcal{N}_1 T_S, \ (\mathcal{N}_1 + \mathcal{N}_2) T_S)$.

According to (45), the value of $\xi(t_{\mathcal{N}_1})$ can be rewritten in the following form

$$\xi(t_{\mathcal{N}_1}) = \begin{bmatrix} \mathbf{q}^L(t_{\mathcal{N}_1}) - \mathcal{T}_{\mathcal{P}_2^L}(t_{\mathcal{N}_1}) \\ \delta_u(t_{\mathcal{N}_1 - 1}) \\ \vdots \\ \delta_u(t_{\mathcal{N}_1 - \bar{d}}) \\ \mathbf{z}(t_{\mathcal{N}_1}) \end{bmatrix} \tag{63}$$

Notice that according to (59), $\mathcal{T}_{\mathcal{P}_2^L}(t_{\mathcal{N}_1}) = [0\,0\,0]^T$. Denote with

$$\mathcal{T}_{\mathcal{P}_1^E}(t_{\mathcal{N}_1}^-) = \lim_{t \to t_{\mathcal{N}_1}^-} \mathcal{T}_{\mathcal{P}_1^E}(t). \tag{64}$$

It represents the desired pose along the trajectory $\mathcal{T}_{\mathcal{P}_1^E}$ before the switch. Let us define the following quantity

$$\mathcal{E}^E(t_{\mathcal{N}_1}^-) = \mathbf{q}^E(t_{\mathcal{N}_1}) - \mathcal{T}_{\mathcal{P}_1^E}(t_{\mathcal{N}_1}^-). \tag{65}$$

It follows that (63) can be rewritten as follows

$$\xi(t_{\mathcal{N}_1}) = \begin{bmatrix} R_E^L \left(\mathcal{E}^E(t_{\mathcal{N}_1}^-) + \mathcal{T}_{\mathcal{P}_1^E}(t_{\mathcal{N}_1}^-) - \mathcal{T}_{\mathcal{P}_2^E}(t_{\mathcal{N}_1}) \right) \\ \delta_u(t_{\mathcal{N}_1 - 1}) \\ \vdots \\ \delta_u(t_{\mathcal{N}_1 - \bar{d}}) \\ \mathbf{z}_{\mathcal{N}_1} \end{bmatrix}. \tag{66}$$

Be $\xi(t_{\mathcal{N}_1}^-)$ the following value

$$\xi(t_{\mathcal{N}_1}^-) = \begin{bmatrix} \mathcal{E}^L(t_{\mathcal{N}_1}^-) \\ \delta_u(t_{\mathcal{N}_1 - 1}) \\ \vdots \\ \delta_u(t_{\mathcal{N}_1 - \bar{d}}) \\ \mathbf{z}(t_{\mathcal{N}_1}) \end{bmatrix}, \tag{67}$$

it is therefore possible to rewrite

$$\xi(t_{\mathcal{N}_1}) = H_E^L \xi(t_{\mathcal{N}_1}^-) + \Pi \tag{68}$$

where

$$H_E^L = \begin{bmatrix} R_E^L & 0^T_{(\bar{d} n_u + n_c) \times n_\mathcal{E}} \\ 0_{(\bar{d} n_u + n_c) \times n_\mathcal{E}} & I \end{bmatrix} \tag{69}$$

and

$$\Pi = [R_E^L \left(\mathcal{T}_{\mathcal{P}_1^E}(t_{\mathcal{N}_1}^-) - \mathcal{T}_{\mathcal{P}_2^E}(t_{\mathcal{N}_1}) \right)^T \ 0^T_{(\bar{d} n_u + n_c)}]^T \tag{70}$$

Remark It is important to note that (70) depends only on the switch between $\mathcal{T}_{\mathcal{P}_1^E}$ and $\mathcal{T}_{\mathcal{P}_2^E}$, so it can be calculated analytically at planning time.

Finally, let $\Gamma_{\mathcal{N}_1}$ be defined as the smallest ellipsoidal set containing $\xi(t_{\overline{\mathcal{N}_1}})$

$$\Gamma_{\mathcal{N}_1} := \left\{\xi : \xi^T P_{\mathcal{N}_1}\xi \leqslant 1,\ P_{\mathcal{N}_1} \geqslant 0\right\}. \tag{71}$$

It can be easily calculated iteratively. Namely, if

$$\xi(t_k) \in \Gamma_k := \left\{\xi : \xi^T P_k \xi \leqslant 1, P_k \geqslant 0\right\},$$

then the ellipsoidal set

$$\Gamma_{k+1} = \left\{\xi : \xi^T P_{k+1}\xi \leqslant 1, P_{k+1} \geqslant 0\right\},$$

so that $\xi(t_{k+1}) \in \Gamma_{k+1}$ can be calculated by solving the following GEVP

$$\min \log |P_{k+1}|^{-1} \tag{72}$$
$$\text{s.t.} \tag{73}$$
$$P_{k+1} = P_{k+1}^T \geqslant 0 \tag{74}$$
$$\alpha_g \geqslant 0,\ g = 1, \cdots, 3 \tag{75}$$
$$1 - \alpha_1 - \alpha_3 \geqslant 0 \tag{76}$$
$$\begin{bmatrix} -\phi^T P_{k+1}\phi & * & * \\ -\alpha_2 C_\sigma^T C_\sigma + \alpha_3 P_k & & \\ -G_\mu^T P_{k+1}\phi & -G_\mu^T P_{k+1} G_\mu & * \\ & +\alpha_1 M_d & \\ -G_p^T P_{k+1}\phi & -G_p^T P_{k+1} G_\mu & -G_p^T P_{k+1} G_p \\ & & +\alpha_2 \end{bmatrix} \geqslant 0 \tag{77}$$

Now, if the following condition holds

$$\Gamma_{\mathcal{N}_1} \oplus \Pi \subseteq \Gamma_0 \tag{78}$$

(62) is definitively fulfilled, we denote by $\oplus$ the Minkowski sum. One way to verify the condition (78) is to preliminarily define

$$\Gamma^*_{\mathcal{N}_1} = \{\xi : \xi^T P^*_{\mathcal{N}_1}\xi \leqslant 1,\ P^*_{\mathcal{N}_1} \geqslant 0\} \tag{79}$$

being the smallest ellipsoidal set for which the condition

$$\Gamma_{\mathcal{N}_1} \oplus \Pi \subseteq \Gamma^*_{\mathcal{N}_1} \tag{80}$$

applies. The set (79), can be calculated by solving the following GEVP problem

$$\min_{\alpha \geqslant 0, P^*_{\mathcal{N}_1} \geqslant 0} \log |P^*_{\mathcal{N}_1}|^{-1} \tag{81}$$
$$\begin{bmatrix} 1 - \alpha - \Pi^T P^*_{\mathcal{N}_1}\Pi & -\Pi^T P^*_{\mathcal{N}_1} H_E^L \\ * & -H_E^{L\,T} P^*_{\mathcal{N}_1} H_E^L + \alpha P_{\mathcal{N}_1} \end{bmatrix} \geqslant 0 \tag{82}$$

The demonstration involving s-procedure arguments [42] has been omitted for the sake of brevity. If the following condition is met

$$\Gamma^*_{\mathcal{N}_1} \subseteq \Gamma_0, \tag{83}$$

the condition (78) is fulfilled.

VI. COORDINATED FEASIBLE TRAJECTORIES PLANNING ALGORITHM

Let Ψ be a discretization by n_Ψ points of the free space $\mathcal{F}$. Let us denote by

$$c = [x_c^E\ y_c^E]^T \in \mathcal{F} \tag{84}$$

the generic point of Ψ expressed in $\mathbf{E}$. Let us now introduce the set

$$\Psi^{(N)} = \Psi \times \Psi \times \Psi \times \cdots \times \Psi. \tag{85}$$

Let us define the weighted oriented graph $\mathcal{G}^{(N)}$ whose nodes correspond to the elements of $\Psi^{(N)}$. Given two generic nodes

$$g_a = \{c_{a_1}^E, c_{a_2}^E, \cdots, c_{a_N}^E\} \in \Psi^{(N)} \tag{86}$$
$$g_b = \{c_{b_1}^E, c_{b_2}^E, \cdots, c_{b_N}^E\} \in \Psi^{(N)}, \tag{87}$$

the arc connecting g_a with g_b is denoted by $\mathcal{W}_{ab}$. Note that $\mathcal{W}_{ab}$ identifies the set of N segments connecting $c_{a_i}^E$ to $c_{b_i}^E$, where $i = 1, \cdots, N$. To restrict the density of the graph $\mathcal{G}^{(N)}$, assume that the arc $\mathcal{W}_{ab}$ only exists if

$$c_{b_i}^E \in \mathcal{S}_{\overline{L}}(c_{a_i}^E),\ i = 1, \cdots, N \tag{88}$$

where $\mathcal{S}_{\overline{L}}(c^E)$ is the largest circle with a radius less than or equal to $\overline{L}$ that is centered in c^E and completely contained in $\mathcal{F}$. Let $c_{I_i}^E(c_{F_i}^E) \in \Psi$ be the initial (final) position for the i-th trajectory. The solution of the planning problem of the N coordinated feasible trajectories starting from $c_{I_i}^E$ and ending in $c_{F_i}^E$, possibly of minimum length, which guarantee the avoidance of collisions between the robots, is thus obtained by recursively exploring the $\mathcal{G}^{(N)}$.

Given a sequence of nodes of $\mathcal{G}^{(N)}$, according to the notation introduced in Sect.V, let $\mathcal{P}^{(N)}$ be a sequence of poses representing the N paths to be traversed at constant forward speed V_D. Consider an SSTMR$_{(i)}$, denote by $\mathcal{T}_{\mathcal{P}_{(i)}^E}$ its trajectory and by $\mathbf{q}_{(i)}^E(t)$ its pose at generic time t. In accordance with Sect. V it is assumed that all trajectories $\mathcal{T}_{\mathcal{P}_{(i)}^E}$ with $i = 1, \cdots, N$ are feasible. According to (13)- (14) it is assumed that the following conditions are checked

$$\mathcal{A}\left(\mathbf{q}_{(i)}^E(t)\right) \in \mathcal{Z}(\mathcal{T}_{\mathcal{P}_{(i)}^E}(t)) \tag{89}$$
$$\mathcal{R}\left(\mathbf{q}_{(i)}^E(t)\right) \subseteq \mathcal{F} \tag{90}$$

and finally, in accordance with (15), given SSTMR$_{(i)}$ and SSTMR$_{(j)}$ with $i, j = 1, \cdots, N$ and $i \neq j$, it is guaranteed that $\forall \mathbf{q}_{(i)}^E(t) : \mathcal{A}\left(\mathbf{q}_{(i)}^E(t)\right) \in \mathcal{Z}(\mathcal{T}_{\mathcal{P}_{(i)}^E}(t))$ and $\forall \mathbf{q}_{(j)}^E(t) : \mathcal{A}\left(\mathbf{q}_{(j)}^E(t)\right) \in \mathcal{Z}(\mathcal{T}_{\mathcal{P}_{(j)}^E}(t))$ with $t \geqslant 0$

$$\mathcal{R}(\mathbf{q}_{(i)}^E(\cdot)) \bigcap \mathcal{R}(\mathbf{q}_{(j)}^E(\cdot)) = \varnothing \tag{91}$$

see Fig. 6. Let $l_{\mathcal{P}^{(N)}} = \sum_{i=1}^N l_{\mathcal{P}_{(i)}^E}$, where $l_{\mathcal{P}_{(i)}^E}$ is the length of $\mathcal{T}_{\mathcal{P}_{(i)}^E}$. The exploration of $\mathcal{G}^{(N)}$ aims to find an alternative sequence of nodes (if any) that identifies a new sequence of

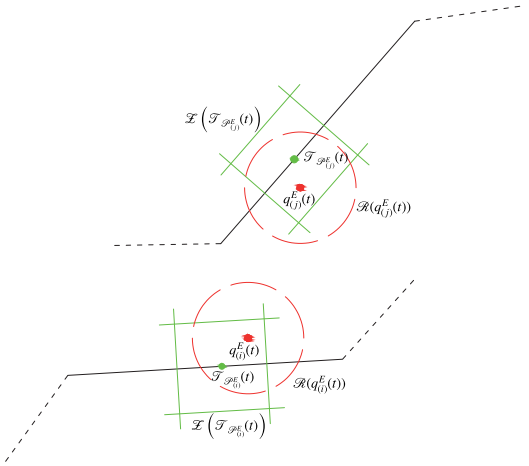

FIGURE 6. The red dots represent the position of the robots. The red circles represent the space occupied by the robots. The green dots represent the points along the trajectories. The green rectangle represents the regions where the robots are guaranteed to be.

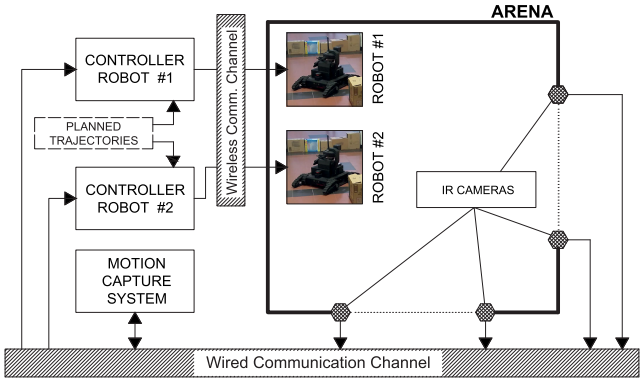

FIGURE 7. Experimental setup architecture.

poses $\overline{\mathcal{P}}^{(N)}$, which is characterized by the same properties of $\mathcal{P}^{(N)}$, but for which the following condition is satisfied

$$l_{\overline{\mathcal{P}}^{(N)}} < l_{\mathcal{P}^{(N)}}.$$

To achieve this, a recursive exploration of the graph $\mathcal{G}^{(N)}$ was chosen in this work, which uses an algorithm of research A*.

VII. RESULTS

In order to demonstrate the effectiveness of the algorithm described in Sect. VI, some planning results obtained for a multi-robot system, involving two robotic platforms, are discussed here. Let us consider the experimental setup depicted in Fig. 7. It consists of two tracked skid-steered robotic units mod. Jaguar v4 by Dr.Robot. The robot has dimensions $0.6 \times 0.7\,\mathrm{m}^2$. Specifically, each robotic platform is connected to its control unit via a common 802.11n data communication channel on a wireless network using the TCP/IP protocol stack. The controllers are responsible for zeroing the trajectory tracking errors, which are estimated using a motion capture system by VICON. The motion capture systems send all relevant information to

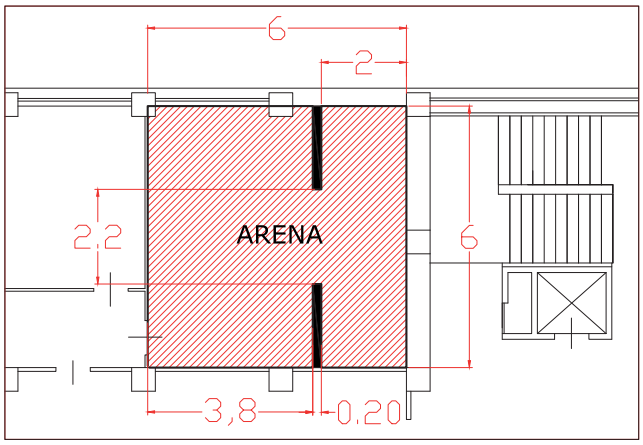

FIGURE 8. Experimental operating scenario.

TABLE 1. Experimental Scenario: relevant values.

Symbol	min	max	typ.	unit
R	-	-	0.08	m
D	-	-	0.5	m
$\mathcal{E}_x$	-0.35	0.35	-	m
$\mathcal{E}_y$	-0.35	0.35	-	m
$\mathcal{E}_\theta$	-60	60	-	deg
$\mathcal{R}$	-	-	0.46	m
V	0	0.4	0.2	m/s
ω	-35	35	0	deg/s
$\mu_r(\mu_l)$	0.75	1.25	1	-
τ	2010^{-3}	28010^{-3}	-	s
T_s	-	-	0.2	s
V_D	-	-	0.2	m/s

the controllers via a wired communication channel using the TCP/IP protocol stack. The pose of the robots is measured synchronously every $T_S = 0.2$ s. The robots operate asynchronously by applying a control action as soon as it is available and maintaining it until a new command (ZOH) is received. In accordance with Sect. IV, the mathematical model proposed in (39)-(42) has been defined. For this purpose, a series of experimental measurements were carried out in advance to estimate both the maximum delay introduced by data communication channels (6) and the allowable values of the sliding coefficients in the operating scenario, see (21). In particular, it is assumed that the sliding coefficients μ_r and μ_l can deviate from the unit nominal value of $\pm 25\%$ and that $\tau_{max} \leqslant 2T_S$. All values required to define the uncertain mathematical model can be found in Tab. 1.

Each robotic unit uses the following discrete-time proportional-integral (PI) control law for trajectory tracking:

$$\mathbf{z}(t_{k+1}) = \mathbf{z}(t_k) + S\tilde{\xi}(t_k)T_S, \tag{92}$$

$$\delta_u(t_k) = K_i\mathbf{z}(t_k) + K_p\tilde{\xi}(t_k). \tag{93}$$

where $\tilde{\xi}(t_k) = \left[\mathcal{E}^T(t_k)\ \delta_u^T(t_{k-1})\ \delta_u^T(t_{k-2})\right]^T$ is the lifted trajectory tracking error (38) required to account for delays introduced by the architecture of control

system and $S = \begin{bmatrix} I_2 & 0_{2\times 5} \end{bmatrix}$,

$$K_i = -\begin{bmatrix} 0.091 & 0 \\ 0 & 0.019 \end{bmatrix}, \tag{94}$$

$$K_p = -\begin{bmatrix} 0.402 & 0.001 & 0 & 0.01 & 0 & 0.025 & 0 \\ 0 & 0.821 & 0.5 & 0 & -0.007 & 0 & 0 \end{bmatrix}. \tag{95}$$

Based on the above information, the mathematical model of the dynamics of the robot's closed-loop trajectory tracking error (46)-(49) was defined. Finally, according to Sect. IV-D the robust ellipsoidal region (55) was calculated. For the planning purposes, the operating environment shown in Fig. 8 with a size of about 6×6 m^2 was considered. Taking into account the robot's dimensions, the circular safety zone has a radius $\mathcal{R} = 0.46$ m, see Sect. II. Based on the maximum permissible errors in trajectory tracking, according to Sect. VI, the region $\mathcal{Z}(0)$, see (13) and (52), takes the following form

$$\mathcal{Z}(0) := \left\{ \{\mathcal{E}_x, \mathcal{E}_y\} \in \mathbf{R}^2 : J \begin{bmatrix} \mathcal{E}_x \\ \mathcal{E}_y \end{bmatrix} \leqslant \mathbf{b} \right\} \tag{96}$$

where

$$J = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix}^T$$

and $\mathbf{b} = \begin{bmatrix} \mathcal{E}_{x,max} & -\mathcal{E}_{x,min} & \mathcal{E}_{y,max} & -\mathcal{E}_{y,min} \end{bmatrix}^T$, see Tab. 1. The free space was discretized, be Ψ the discretization using a regular grid with 515 points with a resolution of 0.25 m. According to Sect. VI, the weighted oriented graph $\mathcal{G}^{(2)}$ was defined by assuming $\overline{L} = 2$ m.

In Figs. 9-10 are shown the results of the planning algorithm. The planning result obtained by applying the method proposed in this paper is denoted by **TP-MRS** and represented by black lines. For the first robot, the identified path consists of 12 segments with a length of 8, 59 m. For the second robot, the identified path consists of 11 segments with a length of 6.26 m. Both paths must be traveled at the prescribed nominal speed $V_D = 0.2$ m/s. As can be seen in Fig. 10, see black solid lines, the collision threshold $2\mathcal{R}$ is never violated and then the (15) is always guaranteed.

Two alternative planning methods, referred to as **TP-D** and **TP-F**, were considered in order to provide a comparison that allows a better evaluation of the results obtained. The method **TP-D**, see red lines in Figs. 9-10, uses Dijkstra's classical algorithm to plan two independent paths connecting given starting and ending points. The **TP-D** does not consider any temporal parameterization of the paths in the planning phase, furthermore the **TP-D** uses an ad-hoc reduced grid of points to account for the constraint (14).

The method **TP-F** uses the set-based approach proposed in [15] to perform the independent planning of two trajectories defined by a sequence of path segments to be traversed at the nominal speed V_D, guaranteeing the fulfillment of the constraints (13)-(14). The results are shown in Figs. 9-10, see green lines.

The results of the comparison are shown in Tab. 2. As expected, the algorithm **TP-D** identifies the two shortest

TABLE 2. Comparison summary table.

	Total Length	Coll. Avoid.	Feasib.	Planning Time
TP-D	24.76 m	No	No	1.89 s
TP-F	25.16 m	No	Yes	23.02 s
TP-MRS	27.78 m	Yes	Yes	107.89 s

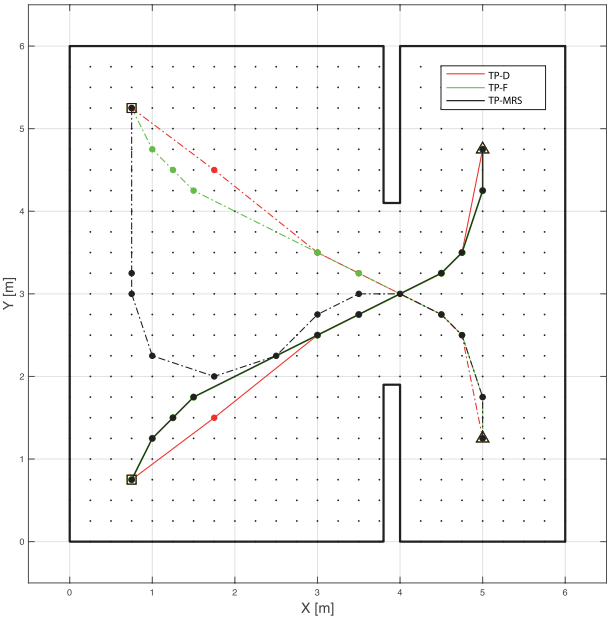

FIGURE 9. The black, red and green solid lines show the sequence of segments composing the planned trajectories. Squares and triangles show the starting and ending points respectively.

trajectories in the shortest time. However, even under nominal conditions (i.e., zero trajectory tracking error) the collision threshold is violated, see red solid line in Fig. 10. The algorithm **TP-F** calculates two independent feasible, non-coordinated trajectories that guarantee a limited tracking error (13). The trajectories calculated with **TP-F** are shorter than those obtained with **TP-MRS** and the planning time is also shorter, yet the collision threshold is violated due to the lack of coordination between the planned trajectories, see green solid line in Fig. 10.

When comparing **TP-D**, **TP-F** and **TP-MRS**, it turns out that the latter is the only way to guarantee the absence of collisions during the robots' movements. As expected, this result is achieved at the price of an increase in the total length of the trajectories combined with a significant increase in computation time.

Figs. 11-12 show the results obtained with 10 experimental runs using the experimental setup described above. These were performed to stress the properties of the trajectories planned by **TP-MRS** through experimental simulations. Fig. 11 shows the trajectories executed by the robots with different starting poses in order to change the initial trajectory tracking errors. Fig. 12 shows the distance between the geometric centers of the robots. The collision threshold is shown in red dashed line. As designed, the distance between

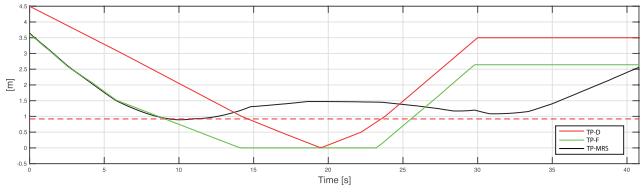

FIGURE 10. Black (TP-MRS) and green (TP-F) solid lines represent the worst-case separation distances between robots. Red (TP-D) solid line represents the distance between the robots under nominal conditions (i.e. zero trajectory tracking error). The red dashed-dot line indicates the collision threshold.

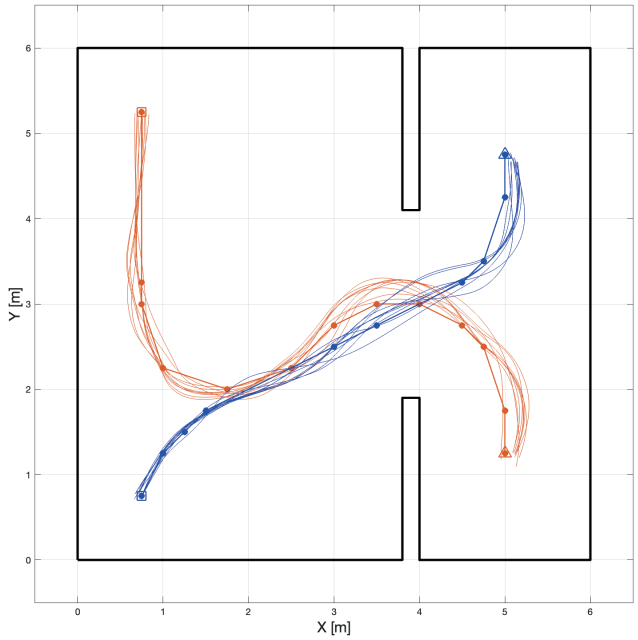

FIGURE 11. Experimental example. The solid orange (blue) lines represent the paths executed by Robot #1 (#2) in different numerical runs at different starting poses.

the robots is always greater than the collision threshold. Figs. 13 and 14 show the trajectory tracking errors and the control velocities of the robots, respectively. As the reader can see, all the values fulfills the relative constraints indicated by the red dashed lines. Finally, to better illustrate the coordinated movement of the two robots, Fig. 15 shows the poses $\mathbf{q}^E_{(1)}$ and $\mathbf{q}^E_{(2)}$ of the two robots at different times during one of the experimental runs performed. The solid green rectangles denote the regions $\mathcal{Z}_{\mathcal{T}_{\mathcal{P}^E_{(1)}}}$ and $\mathcal{Z}_{\mathcal{T}_{\mathcal{P}^E_{(2)}}}$. The red dashed circles represent the circular regions $\mathcal{R}(\mathbf{q}^E_{(1)})$ and $\mathcal{R}(\mathbf{q}^E_{(2)})$ that enclose the footprint of the robots.

VIII. COORDINATED-TRAJECTORIES PLANNING TIME: A NUMERICAL SENSITIVITY ANALYSIS

In order to discuss the computational cost aspect of the proposed algorithm, the results of two numerical examples are presented here. The aim is to stress the computation time aspect of the proposed algorithm for planning coordinated

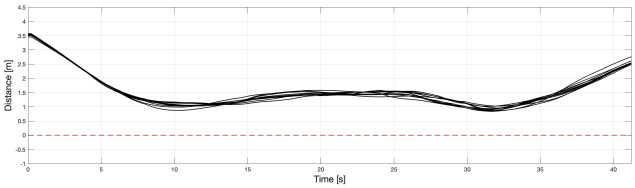

FIGURE 12. Experimental runs. Distance between $\mathcal{R}(\mathbf{q}^E_{(1)})$ and $\mathcal{R}(\mathbf{q}^E_{(2)})$.

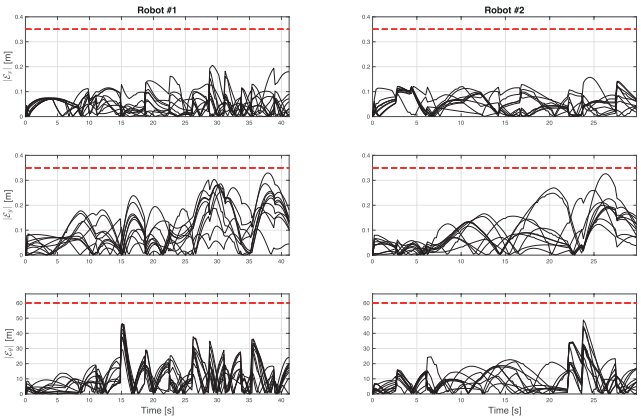

FIGURE 13. Experimental runs. Trajectories tracking errors. Dashed lines represent allowable bounds.

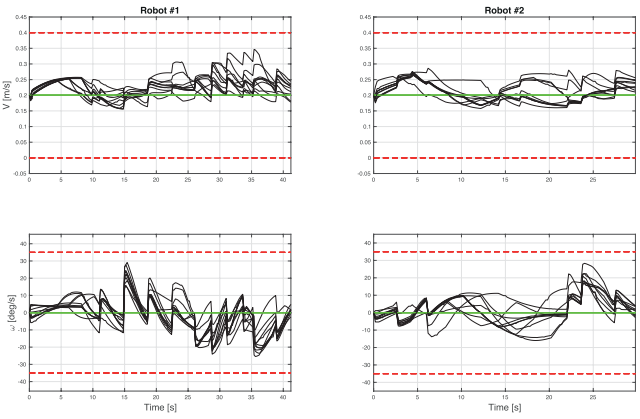

FIGURE 14. Experimental runs. Control speeds. The dashed lines represent allowable bounds. The green solid lines represent the prescribed forward and rotational speeds along the trajectories.

trajectories by varying the numbers of robot units composing the MRS system.

Let's consider the virtual operating environment shown in Fig. 16, discretized with a grid of about 1600 points with a resolution of 0.2 m. The operating environment was divided into 3 regions, each denoted by the sliding coefficients given in Tab 3. For each robotic unit, the trajectory tracking controller (92)-(95) is taken into account. All relevant values can be found in Tab. 1.

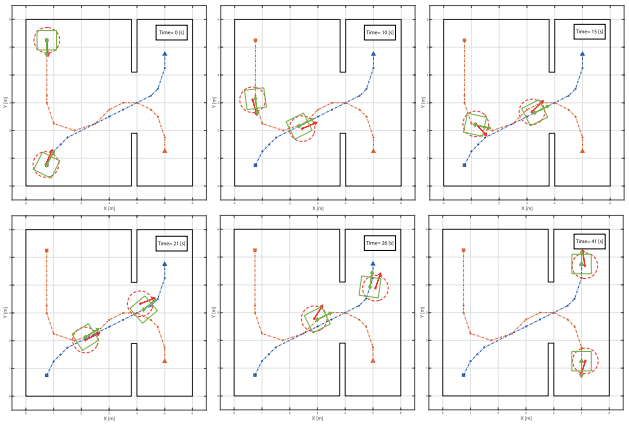

FIGURE 15. Details of an experimental run. Snapshots at different times. The dashed orange (blue) line represents the path taken by Robot #1 (#2). The squares and triangles show the starting and ending points respectively. Solid green rectangles represent the regions $\mathcal{Z}(\mathcal{T}_{\mathcal{PE}_{(1)}})$ and $\mathcal{Z}(\mathcal{T}_{\mathcal{PE}_{(2)}})$, while dashed red circles represent the space occupied by the robots. Green arrows represent $\mathcal{T}_{\mathcal{PE}_{(1)}}$ and $\mathcal{T}_{\mathcal{PE}_{(2)}}$. Red arrows indicate the pose of the robots.

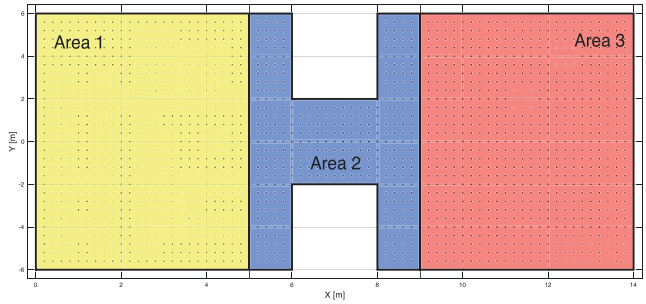

FIGURE 16. Virtual operating environment. The points represent the grid that discretizes the operating environment. Three areas with different sliding coefficients are assumed.

TABLE 3. Numerical simulations: sliding coefficients values.

	Area 1	Area 2	Area 3
Acceleration	$\mu_r = 0.75$ $\mu_l = 0.9$	$\mu_r = 0.95$ $\mu_l = 0.95$	$\mu_r = 0.85$ $\mu_l = 0.75$
Deceleration	$\mu_r = 1.25$ $\mu_l = 1.1$	$\mu_r = 1.05$ $\mu_l = 1.05$	$\mu_r = 1.15$ $\mu_l = 1.25$

A. SCENARIO A

The proposed algorithm was used to find two coordinated and collision-free feasible trajectories that are compatible with the virtual operating environment shown in Fig. 16, take into account skid and slip phenomena due to the sliding coefficients as well as communication delays in the control loop, with minimum length. The result of the coordinated trajectory planning algorithm is shown in Fig. 17 and summarized in Tab. 4. To test the properties of the planned trajectories computed according to the algorithm introduced in Sect. VI, several numerical simulations were performed. Results are summarized in Figs. 18- 21.

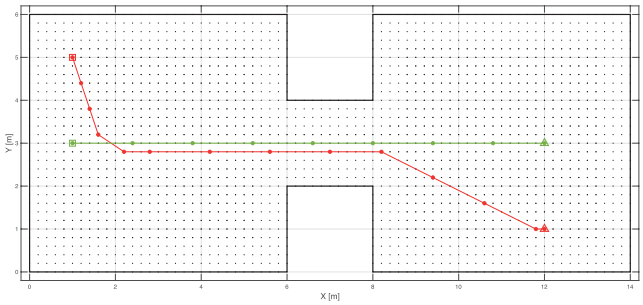

FIGURE 17. Scenario A - Black points represent the grid that discretizes the operating environment. Red and green solid lines show the two coordinated and collision-free feasible trajectories. The filled circles represent the associated waypoints and the squares and triangles indicate the starting and ending points, respectively.

TABLE 4. Scenario A: Trajectory planning result.

	Trajectory data	Planning Time [s]
Robot #1	13 segments, length 12.98 m	363.88
Robot #2	8 segments, length 11 m	

In all numerical simulations performed, a communication delay due to the data communication system was taken into account, which was randomly varied during the simulation with a uniform distribution, and whose maximum and minimum values are compatible with those given in Tab. 1. Furthermore, the initial tracking error was also randomly varied. Fig. 18 shows the trajectories performed by the robots. Fig. 19 shows the tracking errors, which are always below the maximum allowed values, see red dashed line. Fig. 20 shows the forward and rotational speeds of the robots. The considered constraints are highlighted by the red lines. Fig. 21 shows distance between robots. As expected the robustness of the properties of the coordinated and collision-free trajectories, obtained according to the algorithm presented in Sect. VI guarantees that the robots are able to follow the assigned trajectories while respecting the prescribed constraints, despite the presence of delays in the control chain and non-negligible skid or slip phenomena.

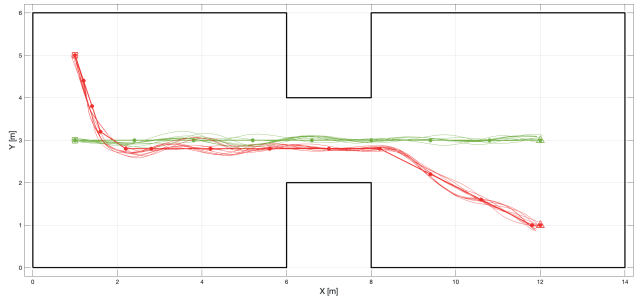

FIGURE 18. Scenario A - Numerical Simulations. The solid red (green) lines represent the trajectories executed by Robot #1 (#2). The squares represent the starting points. The end points are indicated by triangles.

B. SCENARIO B

A scenario involving 3 robotic units is presented here. The operating environment shown in Fig. 16 is considered.

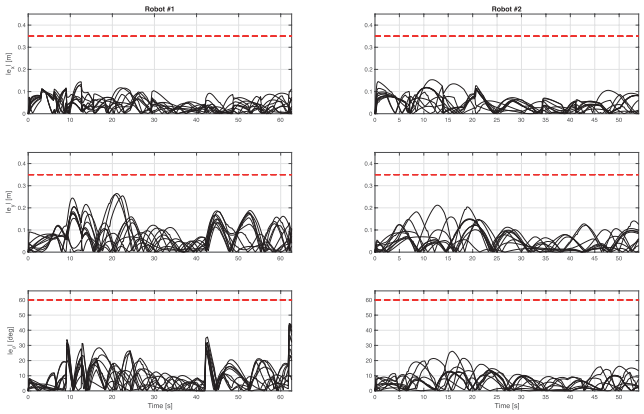

FIGURE 19. Scenario A - Numerical Simulations. Trajectories tracking errors. Dashed lines represent allowable bounds.

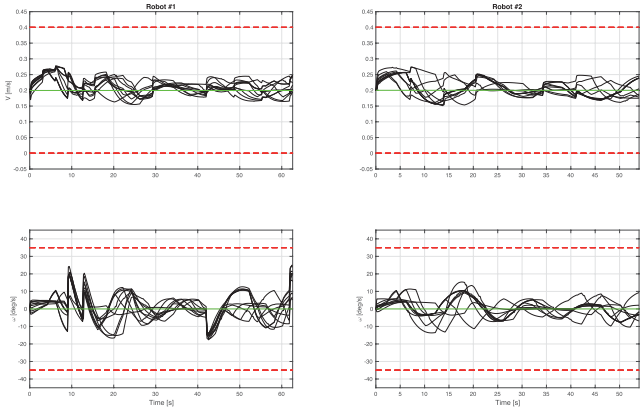

FIGURE 20. Scenario A - Numerical Simulations. Robots' control speeds. Dashed lines represent allowable bounds.

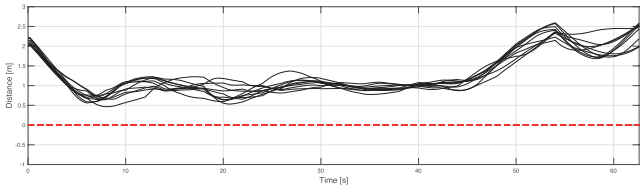

FIGURE 21. Scenario A - Numerical Simulations. Distance between $\mathcal{R}(\mathbf{q}^E_{(1)})$ and $\mathcal{R}(\mathbf{q}^E_{(2)})$.

The result of the coordinated trajectory planning algorithm introduced in Sect. VI is shown in Fig. 22 and summarized in Tab. 5.

To highlight the properties of the planned trajectories obtained by using the algorithm proposed in Sect. VI, several numerical simulations were performed. The obtained results are shown in Figs. 23-26. As well, in all numerical simulations a communication delay due to the data communication system, was considered which was randomly varied with a uniform distribution over time, and whose maximum and minimum values are compatible with those given in Tab. 1. In addition, the initial tracking error was also randomly varied according to the prescribed constraints.

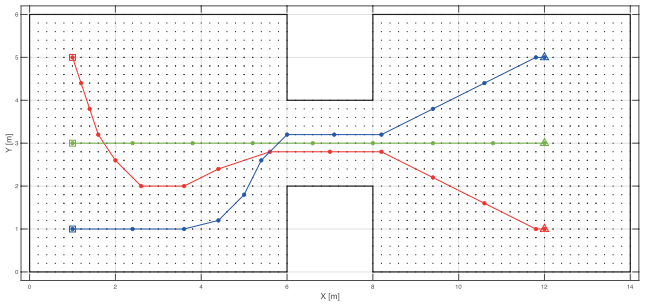

FIGURE 22. Scenario B - Black points represent the grid that discretizes the operating environment. Red, green and blue solid lines represent the three coordinated and collision-free feasible trajectories. The filled circles represent the associated waypoints and the squares and triangles indicate the starting and ending points, respectively.

TABLE 5. Scenario B: Trajectories planning results.

	Trajectory Data	Planning Time [s]
Robot #1	14 segments, length 13.45 m	
Robot #2	8 segments, length 11 m	3860.44
Robot #3	12 segments, length 12.44 m	

Fig. 23 shows the results of numerical simulations. Fig. 24 shows the tracking errors, which are always below the maximum allowed values, see red dashed line. Fig. 25 shows the forward and rotational speeds of the robots. The considered constraints are highlighted by the red lines. Again, the proposed approach ensures that the robots can follow their trajectories without collisions and always comply with the prescribed constraints, despite delays in the control chain and non-negligible skidding or skidding phenomena, and without having to resort to a reactive control algorithm for collision avoidance. Fig. 26 shows distances between robots.

By comparing the Tabs. 4 and 5, a possible weakness of the proposed approach appears: since the search space for optimal trajectories increases with the number of robots, see (85), this has a non-negligible influence on the coordinated trajectories planning time, which may ultimately make this approach unsuitable for scenarios with dense point grids and a large number of robots.

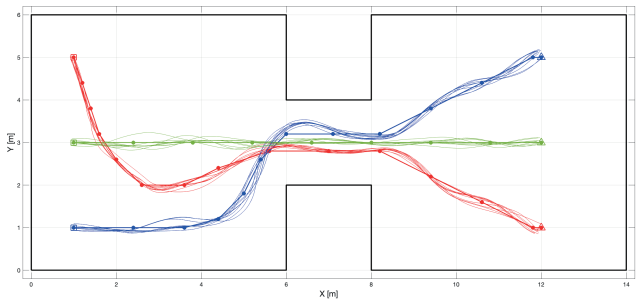

FIGURE 23. Scenario B - Numerical simulations. The solid red, green and blues lines represent the trajectories performed by Robot #1, Robot #2 and Robot #3, respectively. The squares represent the starting points. The end points are indicated by triangles.

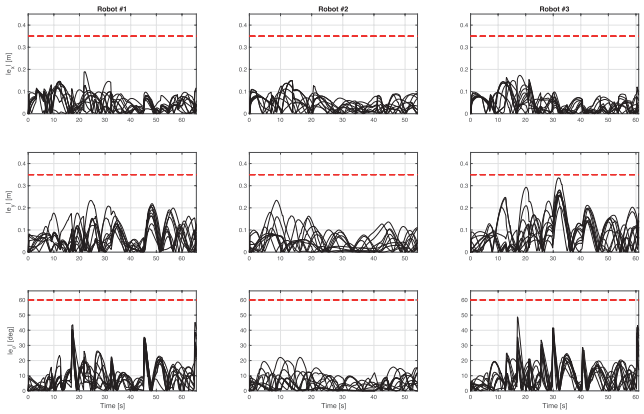

FIGURE 24. Scenario B - Numerical Simulations. Trajectories tracking errors. Dashed lines represent allowable bounds.

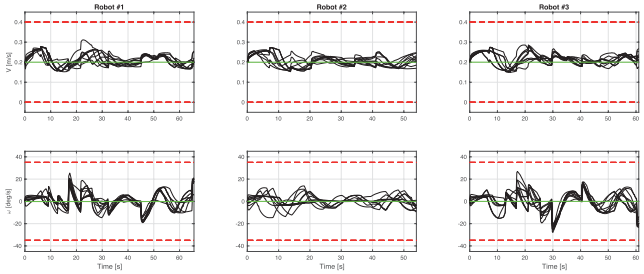

FIGURE 25. Scenario B - Numerical Simulations. Robots' control speeds. Dashed lines represent allowable bounds.

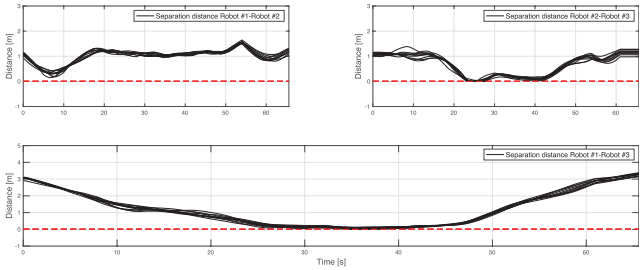

FIGURE 26. Scenario B - Numerical Simulations. Distances between $\mathcal{R}(\mathbf{q}^E_{(1)})$, $\mathcal{R}(\mathbf{q}^E_{(2)})$ and $\mathcal{R}(\mathbf{q}^E_{(3)})$.

IX. CONCLUSION

This paper addresses the problem of planning a set of coordinated trajectories for a group of skid-steered robotic platforms operating in a shared environment. The goal is to define optimal trajectories that ensure collision-free motion of the robots, despite the unavoidable tracking errors caused by limited motion capabilities and external disturbances that limit the authority of the trajectory tracking controllers The trajectories are represented as sequences of path segments, each of which is traversed at a specific nominal speed. The property of the trajectories to guarantee collision-free movements for the robotic units is rigorously guaranteed by considering set-theoretic arguments. The problem of coordinated optimal trajectory planning is reformulated as a graph search combined with a semi-definite programming problem with linear matrix inequalities constraints. The effectiveness of this solution is demonstrated through experiments in a real-world indoor environment involving two remotely-controlled Jaguar V4 robotic units, which encounter communication delays as well as significant skidding and slipping, affecting their handling capabilities. Furthermore, a numerical comparison with a detailed model of the Jaguar V4 is performed to highlight computational burden issues that may limit the applicability of the proposed approach.

Future studies will be devoted to the following topics. On the one hand, efforts will be directed towards reducing the computational burden by investigating the possibility of using alternative graph exploration algorithms that can incorporate set-theoretic constraints formulated through feasibility conditions to guarantee collision-free robot movements. On the other hand, an approach based on the idea of initial delayed feasible trajectories using a set-theoretic methodology will be investigated to guarantee bounded trajectory tracking errors and then collision-free movements.

REFERENCES

[1] K. Masaba and A. Q. Li, "GVGExp: Communication-constrained multi-robot exploration system based on generalized Voronoi graphs," in *Proc. Int. Symp. Multi-Robot Multi-Agent Syst. (MRS)*, Nov. 2021, pp. 146–154.

[2] L. Zhang, Z. Zhang, R. Siegwart, and J. J. Chung, "Distributed PDOP coverage control: Providing large-scale positioning service using a multi-robot system," *IEEE Robot. Autom. Lett.*, vol. 6, no. 2, pp. 2217–2224, Apr. 2021.

[3] N. Bartolini, A. Coletta, and G. Maselli, "On task assignment for early target inspection in squads of aerial drones," in *Proc. IEEE 39th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jul. 2019, pp. 2123–2133.

[4] Z. Yan, N. Jouandeau, and A. A. Cherif, "A survey and analysis of multi-robot coordination," *Int. J. Adv. Robotic Syst.*, vol. 10, no. 12, p. 399, Dec. 2013.

[5] L. Liu, X. Wang, X. Yang, H. Liu, J. Li, and P. Wang, "Path planning techniques for mobile robots: Review and prospect," *Expert Syst. Appl.*, vol. 227, Oct. 2023, Art. no. 120254.

[6] Q. Yao, Y. Tian, Q. Wang, and S. Wang, "Control strategies on path tracking for autonomous vehicle: State of the art and future challenges," *IEEE Access*, vol. 8, pp. 161211–161222, 2020.

[7] P. Hart, N. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Trans. Syst. Sci. Cybern.*, vol. SSC-4, no. 2, pp. 100–107, Feb. 1968.

[8] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Trans. Robot. Autom.*, vol. 12, no. 4, pp. 566–580, Apr. 1996.

[9] S. M. LaValle and J. J. Kuffner, "Randomized kinodynamic planning," *Int. J. Robot. Res.*, vol. 20, no. 5, pp. 378–400, May 2001.

[10] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *Int. J. Robot. Res.*, vol. 30, no. 7, pp. 846–894, Jun. 2011.

[11] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *Autonomous Robot Vehicles*. Cham, Switzerland: Springer, 1986, pp. 396–404.

[12] S. S. Ge and Y. J. Cui, "Dynamic motion planning for mobile robots using potential field method," *Auto. Robots*, vol. 13, no. 3, pp. 207–222, 2002.

[13] J. Chen, W. Zhan, and M. Tomizuka, "Autonomous driving motion planning with constrained iterative LQR," *IEEE Trans. Intell. Vehicles*, vol. 4, no. 2, pp. 244–254, Jun. 2019.

[14] S. Manzinger, C. Pek, and M. Althoff, "Using reachable sets for trajectory planning of automated vehicles," *IEEE Trans. Intell. Vehicles*, vol. 6, no. 2, pp. 232–248, Jun. 2021.

[15] V. Scordamaglia, V. A. Nardi, and A. Ferraro, "A feasible trajectory planning algorithm for a network controlled robot subject to skid and slip phenomena," in *Proc. 24th IEEE Int. Conf. Emerg. Technol. Factory Autom. (ETFA)*, Sep. 2019, pp. 933–940.

[16] V. Scordamaglia, A. Ferraro, F. Tedesco, and G. Franzè, "Set-theoretic approach for autonomous tracked vehicles involved in post-disaster first relief operations," in *Proc. 10th Int. Conf. Control, Decis. Inf. Technol. (CoDIT)*, vol. 28, Jul. 2024, pp. 2006–2011.
[17] X.-M. Hu, Z.-L. Chen, Z. Yang, M. Li, and L. Zhang, "Adaptive set-based CLPSO for path planning of multi-target surveillance with moving obstacles," in *Proc. IEEE 25th Int. Conf. Comput. Supported Cooperat. Work Design (CSCWD)*, May 2022, pp. 1155–1160.
[18] F. Siebenrock, D. Moss, K. Nagatou, M. Schwartz, and S. Hohmann, "Trajectory-planning using set-based motion-primitives considering model uncertainty and controller tracking errors," in *Proc. IEEE 25th Int. Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2022, pp. 2358–2365.
[19] B. Cui, X. Chen, R. Chai, Y. Xia, and H.-S. Shin, "Trajectory planning of spacecraft swarm reconfiguration using reachable set-based collision constraints," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 60, no. 5, pp. 6474–6487, Oct. 2024.
[20] E. Gah, R. Niu, B. Geisel, and S. Z. Yong, "Set-based intent-expressive trajectory planning and intent estimation," *IEEE Control Syst. Lett.*, vol. 7, pp. 151–156, 2023.
[21] X. Qi, D. Theilliol, D. Song, and J. Han, "Invariant-Set-Based planning approach for obstacle avoidance under vehicle dynamic constraints," in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Dec. 2015, pp. 1692–1697.
[22] M. Facerías, V. Puig, and A. Stancu, "Distributed set-based planning in autonomous vehicles," in *Proc. IEEE 29th Int. Conf. Emerg. Technol. Factory Autom. (ETFA)*, vol. 147, Sep. 2024, pp. 1–9.
[23] R. McGovern, N. Athanasopoulos, and S. McLoone, "Safe set-based trajectory planning for robotic manipulators," *IEEE Trans. Robot.*, vol. 40, pp. 3082–3096, 2024.
[24] A. P. Aguiar and J. P. Hespanha, "Trajectory-tracking and path-following of underactuated autonomous vehicles with parametric modeling uncertainty," *IEEE Trans. Autom. Control*, vol. 52, no. 8, pp. 1362–1379, Aug. 2007.
[25] J. Qin and J. Du, "Robust adaptive asymptotic trajectory tracking control for underactuated surface vessels subject to unknown dynamics and input saturation," *J. Mar. Sci. Technol.*, vol. 27, no. 1, pp. 307–319, Mar. 2022.
[26] O. O. Martins, A. A. Adekunle, S. B. Adejuyigbe, O. H. Adeyemi, and M. O. Arowolo, "Wheeled mobile robot path planning and path tracking controller algorithms: A review," *J. Eng. Sci. Technol. Rev.*, vol. 13, no. 3, pp. 152–164, 2020.
[27] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proc. IEEE*, vol. 95, no. 1, pp. 215–233, Jan. 2007.
[28] L. Shi, Z. Ma, S. Yan, and Y. Zhou, "Cucker-smale flocking behavior for multiagent networks with coopetition interactions and communication delays," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 54, no. 9, pp. 5824–5833, Sep. 2024.
[29] W. Li, L. Shi, M. Shi, J. Yue, B. Lin, and K. Qin, "Seeking secure synchronous tracking of networked agent systems subject to antagonistic interactions and denial-of-service attacks," *IEEE Trans. Artif. Intell.*, early access, Dec. 25, 2024, doi: 10.1109/TAI.2024.3522873.
[30] R. C. Arkin, *Behavior-based Robotics*. Cambridge, MA, USA: MIT Press, 1998.
[31] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," *IEEE Trans. Autom. Control*, vol. 62, no. 8, pp. 3861–3876, Aug. 2017.
[32] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robot. Autom. Mag.*, vol. 4, no. 1, pp. 23–33, Mar. 1997.
[33] S. Quinlan and O. Khatib, "Elastic bands: Connecting path planning and control," in *Proc. IEEE Int. Conf. Robot. Autom.*, Jun. 1993, pp. 802–807.
[34] G. Antonelli, F. Arrichiello, and S. Chiaverini, "The null-space-based behavioral control for autonomous robotic systems," *Intell. Service Robot.*, vol. 1, no. 1, pp. 27–39, Jan. 2008.
[35] K. Kant and S. W. Zucker, "Toward efficient trajectory planning: The path-velocity decomposition," *Int. J. Robot. Res.*, vol. 5, no. 3, pp. 72–89, Sep. 1986.
[36] K. Kant and S. W. Zucker, "Planning collision-free trajectories in time-varying environments: A two-level hierarchy," *Vis. Comput.*, vol. 3, no. 5, pp. 304–313, Mar. 1988.
[37] J. Hu, P. Bhowmick, and A. Lanzon, "Group coordinated control of networked mobile robots with applications to object transportation," *IEEE Trans. Veh. Technol.*, vol. 70, no. 8, pp. 8269–8274, Aug. 2021.
[38] G. Muscio, F. Pierri, M. A. Trujillo, E. Cataldi, G. Antonelli, F. Caccavale, A. Viguria, S. Chiaverini, and A. Ollero, "Coordinated control of aerial robotic manipulators: Theory and experiments," *IEEE Trans. Control Syst. Technol.*, vol. 26, no. 4, pp. 1406–1413, Jul. 2018.
[39] Z. Fan, J. Wu, L. Dai, and Y. Xia, "Trajectory planning based on MINVO basis for autonomous vehicles in lane change scenarios," in *Proc. 42nd Chin. Control Conf. (CCC)*, Jul. 2023, pp. 1–8.
[40] G. Borrello, L. Lorusso, M. Basso, and A. Acernese, "Trajectory planning based on model predictive control with dynamic obstacle avoidance in unstructured environments," in *Proc. Eur. Control Conf. (ECC)*, Jun. 2024, pp. 448–455.
[41] W. P. M. H. Heemels, N. van de Wouw, R. H. Gielen, M. C. F. Donkers, L. Hetel, S. Olaru, M. Lazar, J. Daafouz, and S. Niculescu, "Comparison of overapproximation methods for stability analysis of networked control systems," in *Proc. 13th ACM Int. Conf. Hybrid Syst., Comput. Control*, Apr. 2010, pp. 181–190.
[42] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequal. in System and Control Theory*. Philadelphia, PA, USA: SIAM, 1994.
[43] I. Kolmanovsky and E. G. Gilbert, "Theory and computation of disturbance invariant sets for discrete-time linear systems," *Math. Problems Eng.*, vol. 4, no. 4, pp. 317–367, Jan. 1998.

VALERIO SCORDAMAGLIA (Member, IEEE) was born in Italy, in 1977. He received the Ph.D. degree in electrical and automation engineering from the Mediterranea University of Reggio Calabria, Italy, in 2008. Since 2010, he has been with the DIIES Department, Mediterranea University of Reggio Calabria, as a Researcher. His current research interests include constrained predictive control, networked control systems, control under constraints, fault-tolerant systems, and fault detection and isolation algorithms.

ALESSIA FERRARO was born in 1996. She received the master's degree in electronic engineering from the Universitá degli Studi Mediterranea di Reggio, in 2022, where she is currently pursuing the Ph.D. degree. Her research interest includes the development of autonomous robotic systems for first relief after natural disasters. From a theoretical point of view, she has focused on set-theoretic approaches for trajectory planning in multi-robot systems and for fault detection and isolation of autonomous systems.