POLITECNICO DI TORINO
Repository ISTITUZIONALE

Path planning in formation and collision avoidance for multi-agent systems

Original
Path planning in formation and collision avoidance for multi-agent systems / CEN CHENG, P.D., Indri, M., Possieri, C., Sassano, M., Sibona, F.. - In: NONLINEAR ANALYSIS. - ISSN 1751-570X. - STAMPA. - 47:(2023), p. 101293. [10.1016/j.nahs.2022.101293]

Availability:
This version is available at: 11583/2972192 since: 2022-10-12T10:05:40Z

Publisher:
Elsevier

Published
DOI:10.1016/j.nahs.2022.101293

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright
Elsevier postprint/Author's Accepted Manuscript

© 2023. This manuscript version is made available under the CC-BY-NC-ND 4.0 license http://creativecommons.org/licenses/by-nc-nd/4.0/.The final authenticated version is available online at: http://dx.doi.org/10.1016/j.nahs.2022.101293

(Article begins on next page)

19 November 2026

Path planning in formation and collision avoidance for multi-agent systems*

Pangcheng David Cen Cheng[a], Marina Indri[a], Corrado Possieri[b,c], Mario Sassano[b], Fiorella Sibona[a]

[a] *Dipartimento di Elettronica e Telecomunicazioni, Politecnico di Torino, Corso Duca degli Abruzzi, 24, 10129 Torino, Italy*
[b] *Dipartimento di Ingegneria Civile e Ingegneria Informatica, Università di Roma Tor Vergata, Via del Politecnico 1, 00133 Roma, Italy*
[c] *Istituto di Analisi dei Sistemi ed Informatica "A. Ruberti", Consiglio Nazionale delle Ricerche (IASI-CNR), via dei Taurini 19, 00185 Roma, Italy*

Abstract

This paper investigates, in a centralized manner, the motion planning problem for a team of unicycle-like mobile robots in a known environment. In particular, a multi-agent collision-free patrolling and formation control algorithm is presented, which combines outcomes of: (i) stability analysis of hybrid systems, (ii) algebraic geometry, and (iii) classical potential functions. The objective is achieved by designing a Lyapunov-based hybrid strategy that autonomously selects the navigation parameters. Tools borrowed from algebraic geometry are adopted to construct Lyapunov functions that guarantee the convergence to the desired formation and path, while classical potential functions are exploited to avoid collisions among agents and the fixed obstacles within the environment. The proposed navigation algorithm is tested in simulation and then validated by using the robots of a remote accessible robotic testbed.

Keywords: Motion planning, mobile robotics, stability of hybrid systems, Lyapunov methods.

*Part of the material in this paper has been presented at the 2nd IEEE Conference on Control Technology and Applications, held in Copenhagen, Denmark.

Email addresses: pangcheng.cencheng@polito.it (Pangcheng David Cen Cheng), marina.indri@polito.it (Marina Indri), corrado.possieri@iasi.cnr.it (Corrado Possieri), mario.sassano@uniroma2.it (Mario Sassano), fiorella.sibona@polito.it (Fiorella Sibona)

Cite the published paper as:
P.D. Cen Cheng, M. Indri, C. Possieri, M. Sassano, F. Sibona,
"Path planning in formation and collision avoidance for multi-agent systems,"
Nonlinear Analysis: Hybrid Systems, Volume 47, Feb. 2023, 101293
https://doi.org/10.1016/j.nahs.2022.101293

1. Introduction

The use of a group of mobile robots, cooperatively acting toward a common objective, has been significantly growing in the most recent years in several, different applications, from monitoring and target searching in automation and logistics scenarios, to surveillance and rescue missions. The actual advantages deriving from the employment of a robotic team instead of a single mobile agent rely on the possibility of achieving a robust, efficient collaboration among the members of the team, and on their ability of moving in a proper way to carry out the assigned task, avoiding any collision with other elements in the environment and between themselves. The fundamental problems to be addressed from the research point of view are then related to path planning and formation control, to be solved together with collision avoidance.

Multi-robot path planning techniques are aimed at finding the optimal path for the members of the robotic team according to some criteria, e.g., as in [1], where the problem is addressed on graphs over four minimization objectives: the makespan (i.e., the last arrival time), the maximum distance (traveled by the single robot), the total arrival time, and the total distance. Here the computational efficiency of the developed optimization algorithm based on Integer Linear Programming is enhanced through some heuristics to allow handling of hundreds of robots at the expense of slight optimality loss. In [2], a coevolution-based particle swarm optimization method is instead proposed to cope with the multi-robot path planning issue, considering the total path length of the multi-robot system as global objective function. The SplitAndGroup (SAG) algorithm developed in [3] provides constant factor makespan optimal solutions on average over all problem instances on grids and grid-like environments.

Such path planning algorithms, however, do not allow to impose a desired arrangement to the robotic agents while they are moving, as required in some applications, such as patrolling and surveillance. This requisite is addressed by the formation control approaches, which are often based on the definition of suitable navigation functions, originally introduced in [4] using artificial poten-

tial functions for a single robot, and subsequently extended to the multi-robot case (e.g., in [5], [6], [7]). The approaches of this kind exploit the knowledge of the environment topology and of the obstacles present in it to build control policies that guarantee the convergence of the team of robotic agents to the assigned formation scheme, while avoiding collisions [7]. Centralization and decentralization of the control strategy is a key issue: a centralized architecture, including a single control law, can be more complex from the computational point of view, but it can generally guarantee the completeness of the solution. Various decentralized solutions and approaches allowing some kind of scalability have been proposed in literature, e.g., in [6], where the potential function is constructed in a way that facilitates the complete decentralization of the scheme, and in [5], where a decentralized cooperative controller is designed as the gradient of a proper navigation function, whose minimum corresponds to the desired configuration, guaranteeing the obstacle avoidance.

Obviously, computational burden and complexity are more significant aspects in applications involving a large or very large number of robots, for which specific solutions have been proposed in literature, e.g., in [8] and [9]. In [8], the computational burden is reduced through a multi-agent flocking approach in which not all the agents are informed, but only the virtual leader and some of the agents that move with the desired constant velocity; the uninformed agents are able to move with the same desired velocity thanks to periodical interactions with the informed ones. In [9], the problem is addressed introducing an abstraction based on the definition of a map from the robots configuration space to a manifold, whose dimension is lower and independent of the number of robots involved; the approach allows to control the robots formation and the trajectory independently.

The control design is decoupled also in the hierarchical approach proposed in [10] to address the problem of making a group of unicycles converge to a common circle of assigned radius, and travel around it in a desired direction; the information exchange between the mobile agents is modelled by a directed graph, and the control scheme is designed decoupling the problem of making

the unicycles converge to the common circle from the problem of stabilizing the formation. Modelling through formation graphs (i.e., graphs whose nodes capture the individual agent kinematics, and whose edges represent interagent constraints that must be satisfied) is specifically addressed in [11], focusing on the feasibility problem, i.e., the existence of nontrivial agent trajectories that satisfy the interagent constraints, given the kinematics of the agents.

Several formation control approaches are based on leader-follower strategies, mainly with the aim of achieving a simpler and easily scalable architecture, with reduced communication requirements; the downside of this kind of solutions is a potential weakness in practice, if the substitution of the predefined leader is not envisaged or it is even impossible in case of some fault. Among the first theoretical contributions relative to the leader-follower policy, it is worth recalling both the distributed control approach developed in [12], based on artificial potentials and virtual leaders, and the analysis of the stability properties of mobile agent formations based on leader following carried out in [13]. More recent control approaches can be found in [14], [15], [16], addressing specific issues. In particular, the leader-follower approach developed in [14] is focused on formation and tracking control along straight paths; here, the uniform global asymptotic stability of the closed-loop system is guaranteed by partially linear, time-varying controllers with the addition of a nonlinear term, within a strategy in which each robot is a leader to one robot and follower to another, with a unique swarm leader robot that receives the information of the reference trajectory (and a unique tail robot that is leader to none).

The decentralized adaptive formation controller proposed in [15] ensures uniformly ultimate boundedness of the closed-loop system with prescribed transient and steady-state performances, taking into account the presence of external disturbances and uncertainties in the vehicle dynamics; the control objective is to make each vehicle follow its reference trajectory and to avoid collisions between each vehicle and its leader. In [16], prescribed transient and steady-state performances are achieved as well, under communication constraints among the agents, by a decentralized formation control in which only the leader has the

trajectory information; a tan-type barrier Lyapunov function and a recursive adaptive backstepping procedure are adopted to guarantee the asymptotic stability of the closed-loop system.

Further recent contributions can be found in [17] and [18], developed under the consensus protocol; in the first approach, distributed kinematic controllers and neural network torque controllers are derived for each robot to let a group of mobile agents asymptotically converge to a desired geometric pattern along the given reference trajectory, while in the second one the bipartite consensus protocol is adopted under possible communication delays. A quite complete survey of multi-agent formation control approaches can be finally found in [19], where the types of sensed variables used in the different solutions are investigated, classifying the existing schemes into position-displacement and distance-based control approaches.

The centralized approach proposed in this paper solves the patrolling and formation control problems for a group of *unicycle-like* mobile robots, guaranteeing that the trajectories of the multi-robot system are collision-free. Although, differently from the design strategies proposed in the above mentioned references [6]–[19], the approach proposed in this paper is centralized, it possesses several desirable features that are difficult to achieve in a decentralized setting. First, the motion planning architecture given in this paper does not rely on a leader-follower hierarchy and, hence, it is intrinsically robust with respect to faults of one of the robots. Furthermore, the hybrid design proposed to tune online the navigation parameters can be used to optimize the trajectories followed by the team of mobile robots. Such an optimization cannot be easily carried out using a decentralized approach. Finally, this centralized design strategy allows also to guarantee robustness of the navigation scheme with respect to measurement and implementation errors, as formalized in Section 2.

The originality of the contribution is given by the use of a hybrid controller to ensure the convergence to a prescribed set, and the combination of algebraic techniques and methods inspired by the classical navigation functions to achieve the convergence to the desired path in formation, avoiding collisions among the

agents and with static obstacles in the environment. More in detail, the guarantee of convergence provided by the hybrid system approach ensures patrolling in formation, while the adopted barrier function prevents collisions during the transient time. Preliminary results were developed in [20] for the single-robot case and in [21] for the multi-robot case. The single-robot case was also experimentally validated in a laboratory industrial-like environment, considering an offline path planning as first implementation in [22], and then including a procedure for the online upgrade of the computed path in [23].

The main contribution of this work compared to the afore mentioned preliminary results is that, in this paper, the navigation strategies envisioned in [20, 21] are coupled with a hybrid framework that autonomously selects and updates navigation parameters so to guarantee the achievement of the desired formation and patrolling task while avoiding collisions with fixed obstacles in the environment and among agents. Furthermore, in this paper we report all the previously omitted proofs and we carry out a robustness analysis of the proposed navigation strategy, which ensures patrolling in prescribed formation even in the case of inaccurate measurements of the agents' positions and imperfect implementation of the nominal strategy. It is worth to be noted that, thanks to the centralization of the proposed approach, there is no predefined leader among the robots: the desired patrolling paths can be assigned to some or all the agents of the team, thus enhancing the robustness of the scheme.

In this paper, the validity of the approach is proved for groups of three or four mobile agents, both in simulation and in experimental tests, carried out using a remotely accessible robotic testbed (namely, `Robotarium` [24]). The results achieved in all the tests confirm the effectiveness of the proposed solution, which is expected to be applicable up to ten robots, or even more for simple agents like the ones of the remote testbed. Clearly, the scalability of the approach is related to the computational and communication potentialities of the available hardware/software architecture.

1.1. Notation

In this section, we introduce the notation used all throughout the paper and we formalize the considered problem. Let $\mathbb{N}$, $\mathbb{R}$, $\mathbb{R}_{\geqslant 0}$, and $\mathbb{R}_{>0}$ denote the sets of natural, real, nonnegative real, and positive real numbers, respectively. The symbol I_n denotes the n-dimensional identity matrix, whereas the symbol $\mathrm{diag}(a_1, \ldots, a_n)$ denotes the diagonal matrix whose diagonal entries starting in the upper left corner are $a_1, \ldots, a_n$. Given a function $V : \mathbb{R}^n \to \mathbb{R}$, the symbol ∇V denotes its gradient. Given two vectors $v, w \in \mathbb{R}^n$, the symbol $\langle v, w \rangle$ denotes their inner product. Given a set $\mathcal{V} \subset \mathbb{R}^n$ and a vector $x \in \mathbb{R}^n$, let $\|x\|_{\mathcal{V}} := \inf_{y \in \mathcal{V}} \|x - y\|$. The symbol $\mathcal{B}$ denotes the closed unit ball in the Euclidean norm of appropriate dimensions. Let $\mathrm{col}(x, y) := [\ x^\top \quad y^\top\ ]^\top$. The symbols $f : \mathbb{R}^n \to \mathbb{R}^n$ and $F : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ denote functions and set-valued mappings from $\mathbb{R}^n$ to $\mathbb{R}^n$, respectively. A function $\alpha : \mathbb{R}_{\geqslant 0} \to \mathbb{R}_{\geqslant 0}$ is of class $\mathcal{K}_\infty$, denoted as $\alpha \in \mathcal{K}_\infty$, if it is continuous, zero at zero, strictly increasing, and unbounded. A function $\psi : \mathbb{R}_{\geqslant 0} \times \mathbb{R}_{\geqslant 0} \to \mathbb{R}_{\geqslant 0}$ is of class $\mathcal{KL}$, denoted as $\psi \in \mathcal{KL}$, if it is nondecreasing in its first argument with $\lim_{r \to 0} \psi(r, t) = 0$ for each $t \in \mathbb{R}_{\geqslant 0}$ and it is nonincreasing in its second argument with $\lim_{t \to +\infty} \psi(r, t) = 0$ for all $r \in \mathbb{R}_{\geqslant 0}$. Given a set $\mathcal{V} \subset \mathbb{R}^n$, a function $\varrho : \mathbb{R}^n \to \mathbb{R}_{\geqslant 0}$ is positive definite with respect to $\mathcal{V}$, denoted as $\varrho \in \mathcal{PD}(\mathcal{V})$, if $\varrho(x) = 0$ for all $x \in \mathcal{V}$ and $\varrho(x) > 0$ for all $x \in \mathbb{R}^n \setminus \mathcal{V}$. The ring of all the polynomials in x with real coefficients is denoted as $\mathbb{R}[x]$. Given $p_1, \ldots, p_\varkappa \in \mathbb{R}[x]$ the variety of $p_1, \ldots, p_\varkappa$ is $\mathbf{V}(p_1, \ldots, p_\varkappa) := \{x \in \mathbb{R}^n : p_i(x) = 0,\, i = 1, \ldots, \varkappa\}$. When dealing with hybrid systems, we use the same notation and nomenclature of [25], whereas when dealing with algebraic geometry concepts, we use the notation of [26, 27]. As an example, given $q \in \mathbb{R}^{n_1 \times n_2}[x]$, the symbol $\mathrm{Syz}(q)$ denotes a presentation matrix for the set of all the polynomial vectors $p \in \mathbb{R}^{n_2}[x]$ such that $q\, p = 0$.

1.2. Problem formulation

In this paper, the interest lies in designing paths of motion for *unicycle-like* mobile robots moving on the Euclidean plane (see Fig. 1), described by

equations of the form [28]

$$\dot{X} = \cos(\Theta)v, \tag{1a}$$

$$\dot{Y} = \sin(\Theta)v, \tag{1b}$$

$$\dot{\Theta} = \omega, \tag{1c}$$

where $(X(t), Y(t)) \in \mathbb{R}^2$ denotes the Cartesian position of the center of mass, $\Theta(t) \in \mathbb{R}$ denotes the orientation with respect to the horizontal axis, $v(t) \in \mathbb{R}$ and $\omega(t) \in \mathbb{R}$ represent the linear and the angular velocity inputs, respectively.

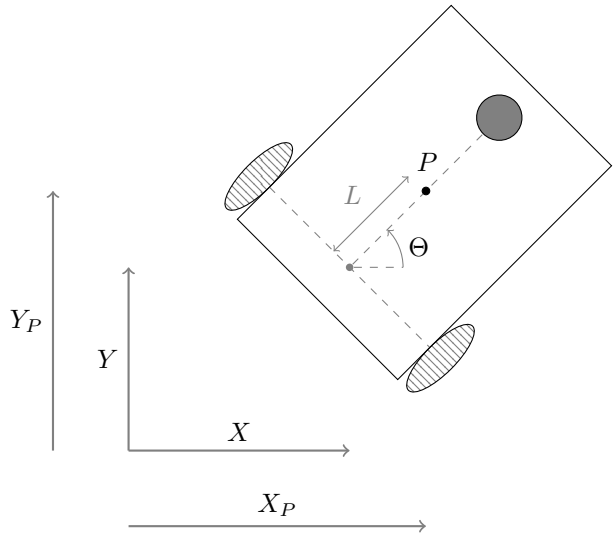


Figure 1: Schematic representation of a unicycle-like mobile robot. The hatched areas represent the actuated wheels of the mobile robot, the filled gray area represents its passive wheel, the point P is the center of mass of the robot, and the length L is the distance of such a point to the line connecting the two wheels. In this two-wheels configuration, the input v represent the mean velocity of the two wheels whereas the input ω represents their differential velocity.

By hinging upon well-known techniques for instance illustrated in [29, 30] and recalled in [20], the relative dynamics of any point on the robot that does not belong to the segment connecting the two wheels (referred to as P in Fig. 1) can be feedback linearized and consequently arbitrarily assigned. Namely, the

dynamics of the Cartesian position $(X_P, Y_P) \in \mathbb{R}^2$ of the point P are given by

$$\dot{X}_P = v\cos(\Theta) - L\omega\sin(\Theta),$$
$$\dot{Y}_P = v\sin(\Theta) + L\omega\cos(\Theta).$$

In the following, we select P as the center of mass of the mobile robot.

Thus, assuming that $L \neq 0$, i.e., that the center of mass of the mobile robot does not belong to the line connecting the two wheels, and letting

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} \cos(\Theta) & \sin(\Theta) \\ -\dfrac{\sin(\Theta)}{L} & \dfrac{\cos(\Theta)}{L} \end{bmatrix} \begin{bmatrix} u_X \\ u_Y \end{bmatrix}, \tag{2}$$

where $u = \begin{bmatrix} u_X & u_Y \end{bmatrix}^\top$ is an auxiliary input, the dynamics of the point P are described by *virtual* single integrators of the form

$$\dot{X}_P = u_X, \tag{3a}$$
$$\dot{Y}_P = u_Y. \tag{3b}$$

Therefore, letting $x_i = \begin{bmatrix} X_{P,i} & Y_{P,i} \end{bmatrix}^\top$, $i = 1, \ldots, N$, denote the position of the center of mass of the i–th mobile robot and letting its inputs v_i and ω_i be

$$\begin{bmatrix} v_i \\ \omega_i \end{bmatrix} = \begin{bmatrix} \cos(\Theta_i) & \sin(\Theta_i) \\ -\dfrac{\sin(\Theta_i)}{L_i} & \dfrac{\cos(\Theta_i)}{L_i} \end{bmatrix} u_i, \tag{4}$$

the main objective of this paper is to design the input u_i so to guarantee that the corresponding solution constitutes a collision free path for the i–th agent, with $i = 1, \ldots, N$, and the set of all solutions steers the mobile robots to patrol a preassigned path in controlled formation.

This problem is addressed by combining:

- A hybrid controller that autonomously tunes the weights of a parametric continuous-time system to ensure convergence to a prescribed set (see Section 2).

- An algebraic technique that allows us to design parametric vector fields such that the solution to the associated dynamical system achieves patrolling, formation control and obstacle avoidance (see Section 3).

The rest of the paper is organized as follows. The Lyapunov-based hybrid strategy that allows to autonomously tune parameters to guarantee convergence of the system is outlined in Section 2. The design of a vector field to achieve patrolling and formation control of the multi-agent system with obstacle avoidance is described in Section 3. Numerical simulations demonstrating the effectiveness of the proposed approach are provided in Section 4. The applicability of the strategy is then demonstrated by experimental tests, whose results are showcased in Section 5. Finally, conclusions are drawn in Section 6. Figure 2 depicts a conceptual scheme summarizing the relations among the topics of each section.

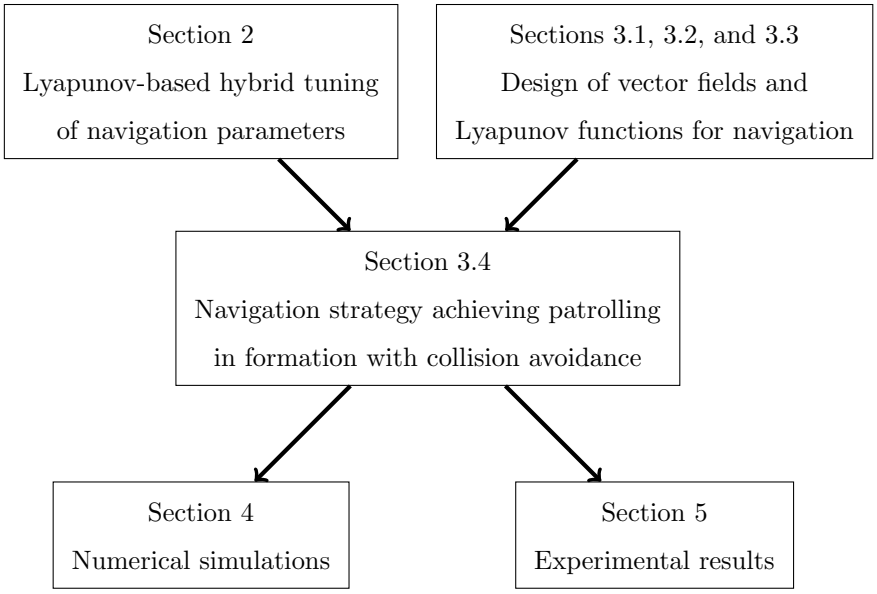


Figure 2: Conceptual scheme of the relations among the topics of each section.

2. Hybrid stabilization of parametric continuous-time systems

The main goal of this section is to propose a novel hybrid strategy that autonomously tunes the parameters of a parametric vector field to ensure that a given set is asymptotically stable for the corresponding dynamical system. Such a strategy is coupled in the subsequent Section 3.4 with an algebraic geometry

technique, which allows to jointly design parametric vector fields having a given variety as attractive and invariant set and the corresponding Lyapunov function to solve the motion planning in formation with collision avoidance problem.

Consider the parametric nonlinear system

$$\dot{x} = f(x, k), \tag{5}$$

where $x(t) \in \mathbb{R}^n$ denotes the state, $k \in \mathcal{A}$ is a vector of parameters, $\mathcal{A} \subset \mathbb{R}^m$ is the compact set of admissible values for the parameters, and the mapping $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ is assumed to be C^z for some sufficiently large $z \in \mathbb{N}$.

Definition 1. A compact set $\mathcal{V} \subset \mathbb{R}^n$ is *locally asymptotically stabilizable for* (5) if there exist $k^\circ \in \mathcal{A}$, an open set $\mathcal{W} \subset \mathbb{R}^n$ containing $\mathcal{V}$, a function $V \in \mathrm{C}^1$, functions $\alpha_1, \alpha_2 \in \mathcal{K}_\infty$, and a function $\varrho \in \mathcal{PD}(\mathcal{V})$ such that, for all $x \in \mathcal{W}$,

$$\alpha_1(\|x\|_{\mathcal{V}}) \leqslant V(x) \leqslant \alpha_2(\|x\|_{\mathcal{V}}), \tag{6a}$$

$$\langle \nabla V(x), f(x, k^\circ) \rangle \leqslant -\varrho(x). \tag{6b}$$

By classical Lyapunov arguments [31], if (6) holds, then the set $\mathcal{V}$ is locally asymptotically stable for the system $\dot{x} = f(x, k^\circ)$. Therefore, if the auxiliary inputs $u_1, \ldots, u_N$ appearing in (4) are designed as $\begin{bmatrix} u_1 & \cdots & u_N \end{bmatrix} = f(x, k^\circ)$, where $x = \begin{bmatrix} X_{P,1} & Y_{P,1} & \cdots & X_{P,N} & Y_{P,N} \end{bmatrix}^\top$, and the set $\mathcal{V}$ describes the target patrolling curve in formation, then the inputs $v_1, \omega_1, \ldots, v_N, \omega_N$ constitute a local *centralized* solution to the patrolling in formation problem for the considered team of mobile robots.

In this section, following constructions similar to those given in [32], we discuss a control design technique that guarantees global asymptotic stability of the set $\mathcal{V}$ for a hybrid implementation of system (5), under the following Assumptions 1 and 2.

Assumption 1. Set $\mathcal{V}$ is locally asymptotically stabilizable for (5) and functions V and ϱ such that (6) holds are given.

A technique to design functions V and ϱ such that (6) holds, hence ensuring that the set $\mathcal{V}$ is locally asymptotically stabilizable for (5), is proposed in the subsequent Section 3.

In order to streamline the exposition of our results, let

$$\ell(x,k) := \langle \nabla V(x), f(x,k) \rangle, \tag{7}$$

and, given a parameter $\mu \in (0,1)$, define the sets

$$\mathcal{C} := \{(x,k) \in \mathbb{R}^n \times \mathcal{A} : \ell(x,k) \leqslant -\mu\varrho(x)\}, \tag{8a}$$

$$\mathcal{D} := \{(x,k) \in \mathbb{R}^n \times \mathcal{A} : \ell(x,k) \geqslant -\mu\varrho(x)\}. \tag{8b}$$

Then, the proposed condition for stabilization of $\mathcal{V}$ via hybrid implementation of system (5) can be stated as follows.

Assumption 2. For all $x \in \mathcal{U} \subset \mathbb{R}^n$ there is $\kappa \in \mathcal{A}$ such that

$$\ell(x,\kappa) \leqslant -\varrho(x). \tag{9}$$

Note that under Assumption 1, it results that $\mathcal{W} \subset \mathcal{U}$. Hence, consider the *hybrid implementation of system* (5) given by the following hybrid system

$$(x,k) \in \mathcal{C}, \quad \begin{cases} \dot{x} = f(x,k), \\ \dot{k} = 0, \end{cases} \tag{10a}$$

$$(x,k) \in \mathcal{D}, \quad \begin{cases} x^+ = x, \\ k^+ \in \operatorname{argmin}_{\kappa \in \Xi(x,k)} \Pi(x,k,\kappa), \end{cases} \tag{10b}$$

where $\Xi(x,k)$ is defined for each $(x,k) \in \mathbb{R}^n \times \mathcal{A}$ as

$$\Xi(x,k) := \{\kappa \in \mathcal{A} : \ell(x,\kappa) \leqslant -\varrho(x)\},$$

and $\Pi(x,k,\kappa)$ is a lower semicontinuous function introduced to systematically select the most desirable k^+ according to some optimality criterion (*e.g.*, minimum norm, minimum deviation from the current k, and so on); see [32] for a discussion on the relationship between the hybrid implementation (10) and the techniques given in [33, 34, 35]. It should be noted that under Assumption 2, the set $\Xi(x,k)$ is nonempty for all $(x,k) \in \mathcal{U} \times \mathcal{A}$. Hence, under the hypothesis that $\mathcal{U} = \mathbb{R}^n$, in the following theorem (whose proof is given in Appendix A), we show that the set $\mathcal{V} \times \mathcal{A}$ is globally asymptotically stable for the hybrid implementation (10).

Theorem 1. *Let the lower semicontinuous function* $\Pi(x,k,\kappa)$ *be level bounded in* κ *locally uniformly in* (x,k) *and let Assumptions 1 and 2 hold with* $\mathcal{U}=\mathbb{R}^n$. *Define* $\varpi(x,k):=\inf_{\kappa\in\Xi(x,k)}\Pi(x,k,\kappa)$ *and assume that it is locally bounded and continuous. Then the set* $\mathcal{V}\times\mathcal{A}$ *is globally asymptotically stable for the hybrid implementation* (10).

By weakening the assumptions of Theorem 1, we can still guarantee uniform local asymptotic stability of the set $\mathcal{V}$, as stated in the following corollary, whose proof is wholly similar to the one of [32, Cor. 1] and hence is omitted.

Corollary 1. *Let the lower semicontinuous function* $\Pi(x,k,\kappa)$ *be level bounded in* κ *locally uniformly in* (x,k) *and let Assumptions 1 and 2 hold for some* $\mathcal{U}\subset\mathbb{R}^n$. *Define* $\varpi(x,k):=\inf_{\kappa\in\Xi(x,k)}\Pi(x,k,\kappa)$ *and assume that it is locally bounded from above and* C^0. *Then the set* $\mathcal{V}\times\mathcal{A}$ *is locally asymptotically stable for the hybrid implementation* (10).

In view of Corollary 1, the main interest in the hybrid implementation (10) relies on the fact that the estimate of the basin of attraction of $\mathcal{V}$ for system

$$\dot{x}=f(x,k^\circ)$$

obtained by using V as Lyapunov function is a subset of the one for the hybrid implementation (10) since $\mathcal{W}\subset\mathcal{U}$.

The next remark provides insights on the solution to (10).

Remark 1. Differently from [32], solutions to the hybrid system (10) need not be eventually continuous. This is due to the fact that the state k of system (10) need not converge to the value k° satisfying (6), and hence the state of system (10) may persistently jump approaching $\mathcal{V}\times\mathcal{A}$. Nonetheless, if either

$$\ell(x,k)\leqslant-\varrho(x),\quad\forall(x,k)\in\mathcal{W}\times\mathcal{K},$$

or $\Pi(x,k,\kappa)$ is designed so that $\operatorname{argmin}_{\kappa\in\Xi(x,k)}\Pi(x,k,\kappa)=k^\circ$ for all $(x,k)\in\mathcal{W}\times\mathcal{K}$, then, since the flow set is eventually positively invariant, solutions $x(t,j)$ to system (10) such that $\{(t,j)\in\operatorname{dom}(x):x(t,j)\in\mathcal{V}\}=\emptyset$ have a semi-global uniform dwell-time and are eventually continuous.

The next remark suggests a selection for the function Π.

Remark 2. A possible (trivial) selection for the function Π in (10b) that meets the hypotheses of Theorem 1 and Corollary 1 is $\Pi(x,k,\kappa)=1$ for all $(x,k,\kappa)\in\mathbb{R}^n\times\mathbb{R}^m\times\mathbb{R}^m$. In such a case, the jump map (10b) can be simply rewritten as

$$(x,k)\in\mathcal{D},\quad\begin{cases}x^+=x,\\ k^+\in\Xi(x,k).\end{cases}$$

It is worth pointing out that Theorem 1 and Corollary 1 establish robustness of local asymptotic stability of the set $\mathcal{V}\times\mathcal{A}$ for system (10). Namely, under the hypotheses of Corollary 1, by [25, Thm. 7.21] and the proof of Theorem 1, since the hybrid system (10) is well-posed and the set $\mathcal{V}\times\mathcal{A}$ is locally asymptotically stable, letting $\mathcal{I}_{\mathcal{V}\times\mathcal{A}}$ be its basin of attraction, then system (10) is robustly $\mathcal{KL}$ pre-asymptotically stable on $\mathcal{I}_{\mathcal{V}\times\mathcal{A}}$; see [25, Def. 6.27, 7.3, 7.10 and 7.18]. In particular, there exists a function $\varphi:\mathbb{R}^n\times\mathbb{R}^p\to\mathbb{R}_{\geqslant 0}$ that is positive on $\mathcal{I}_{\mathcal{V}}\setminus(\mathcal{V}\times\mathcal{A})$ such that $\mathcal{V}\times\mathcal{A}$ is $\mathcal{KL}$ pre-asymptotically stable (see [25, Thm. 3.40] for the characterization of asymptotic stability via $\mathcal{KL}$ functions) on $\mathcal{I}_{\mathcal{V}\times\mathcal{A}}$ for the system

$$(x,k)\in\mathcal{C},\quad\begin{cases}\dot{x}\in F_\varphi(x,k),\\ \dot{k}=0,\end{cases}\tag{11a}$$

$$(x,k)\in\mathcal{D},\quad\begin{cases}x^+=x,\\ k^+\in\operatorname{argmin}_{\kappa\in\Xi(x,k)}\Pi(x,k,\kappa),\end{cases}\tag{11b}$$

where, letting $\xi=\operatorname{col}(x,k)$, the inflated set-valued map $F_\varphi:\mathbb{R}^n\times\mathbb{R}^m\rightrightarrows\mathbb{R}^n$ is given by

$$F_\varphi(\xi):=\overline{\operatorname{con}}(f((\xi+\varphi(\xi)\mathcal{B})\cap\mathcal{C}))+\varphi(\xi)\mathcal{B}.$$

Note that the inflated set-valued map F_φ accounts for both measurement error (through the term $\xi+\varphi(\xi)\mathcal{B}$ at argument of f) and implementation error (through the additive term $+\varphi(\xi)\mathcal{B}$). Hence, this robustness is particularly desirable when dealing with mobile robots, where position and actuation errors are unavoidable in realistic scenarios.

Furthermore, since $\mathcal{C} \cup \mathcal{D} = \mathbb{R}^n \times \mathcal{A}$ and $k(0,0) \in \mathcal{A}$ implies $k(t,j) \in \mathcal{A}$ for all $(t,j) \in \mathrm{dom}(k)$, trajectories of system (11) are complete. Therefore, the set $\mathcal{V} \times \mathcal{A}$ is $\mathcal{KL}$ asymptotically stable on $\mathcal{I}_{\mathcal{V}\times\mathcal{A}}$ for the inflated system (11), that is there is a function $\psi \in \mathcal{KL}$ such that every solution $\xi = \mathrm{col}(x,k)$ to system (11) satisfies, for all $(t,j) \in \mathrm{dom}(\xi)$,

$$\|\xi(t,j)\|_{\mathcal{V}\times\mathcal{A}}(\|\xi(t,j)\|_{\mathbb{R}^{n+m}\setminus\mathcal{I}_{\mathcal{V}\times\mathcal{A}}})^{-1} \leqslant \psi(\|\xi(0,0)\|_{\mathcal{V}\times\mathcal{A}}(\|\xi(0,0)\|_{\mathbb{R}^{n+m}\setminus\mathcal{I}_{\mathcal{V}\times\mathcal{A}}})^{-1}, t+j).$$

3. Design of a vector field to obtain patrolling, formation control and obstacle avoidance

The main goal of this section is the design of an algorithmic procedure to compute a parametric vector field $f(x,k)$, together with functions V and ϱ satisfying Assumptions 1 and 2, such that the solutions to system (5) constitute a path for mobile robots that is collision-free and that steers them to patrol a preassigned path in controlled formation. This goal is pursued by combining algorithms that use tools borrowed from algebraic geometry (see Sections 3.1 and 3.2) with methods inspired by classical navigation functions (see Section 3.3). The former allow to automatically construct Lyapunov functions certifying the convergence to the desired path in formation, in the absence of obstacles, while the latter allow to avoid collisions among agents and with fixed obstacles.

3.1. Review on attractive affine varieties

In this section, we briefly recall some results given in [29, 36, 37] and partially summarized also in [20, 21], which allow to systematically design a vector field $h(x)$ whose associated dynamical system $\dot{x} = h(x)$, has a given affine variety $\mathcal{V} \subset \mathbb{R}^n$ as attractive and h-invariant set. Given $p_1, \ldots, p_s \in \mathbb{R}[x]$, the following Algorithm 1, taken from [36, 37], uses some tools borrowed from algebraic geometry to find a set of vector fields $h \in \mathbb{R}^n[x]$ such that $\mathcal{V} := \mathbf{V}(p_1, \ldots, p_s)$ is

attractive and invariant for $\dot{x} = h(x)$ (we refer the reader to [37] for the computational details of such an algorithm and for the feasibility of its solutions).

Algorithm 1

Input: $p_1, \ldots, p_s \in \mathbb{R}[x]$ such that $\mathcal{V} = \mathbf{V}(p_1, \ldots, p_s)$.

Output: a class of vector fields $h \in \mathbb{R}^n[x]$ and a positively invariant set $\mathcal{I}_\mathcal{V} \subset \mathbb{R}^n$ such that $\mathcal{V}$ is h-invariant and attractive in $\mathcal{I}_\mathcal{V}$.

1: Compute the reduced Gröbner basis $\mathcal{G} = \{\theta_1, \ldots, \theta_\ell\}$ of the sub-module $\langle \nabla p \rangle : \langle p \rangle$.

2: Select $\lambda_i \in (\langle \nabla p \rangle : \langle p \rangle)$, $i = 1, \ldots, n$, such that, letting $\Lambda = \begin{bmatrix} \lambda_1 & \cdots & \lambda_n \end{bmatrix}$, the matrix $\Lambda + \Lambda^\top$ is positive semidefinite.

3: Let $c > 0$ be such that the polynomial matrix Λ is positive definite in $\mathcal{I}_\mathcal{V} := \{x \in \mathbb{R}^n : \sum_{i=1}^s p_i^2(x) < c\}$.

4: Solve the polynomial equation $\langle \nabla p, h \rangle + \Lambda p = 0$ in h.

5: **return** h and $\mathcal{I}_\mathcal{V}$.

In the following sections, firstly an ideal obstacle-free scenario is considered and Algorithm 1 is employed to ensure patrolling of a desired path having the robots maintaining a specific formation among them (Section 3.2), and then the obtained vector fields are modified in order to avoid collisions with obstacles and among the agents (Section 3.3).

3.2. Motion planning and formation control

Let $x_i = \begin{bmatrix} x_{i,1} & x_{i,2} \end{bmatrix}^\top$ denote the Cartesian position of the i–th agent, $i = 1, \ldots, N$, let $n = 2N$, and let $x = \begin{bmatrix} x_1^\top & \cdots & x_N^\top \end{bmatrix}^\top$ denote the adjoint state of the *multi-agent model*. Thus, let $p_{i,j} \in \mathbb{R}[x_i]$, $i = 1, \ldots, s_i$, where s_i is a positive integer, be such that the affine variety

$$\mathcal{V}_i = \mathbf{V}(p_{i,1}, \ldots, p_{i,s_i}) \subset \mathbb{R}^2 \tag{12}$$

is the desired patrolling path for the i–th agent. Furthermore, let $q_1, \ldots, q_\omega \in \mathbb{R}[x]$, where ω is a positive integer, be such that the affine variety

$$\mathcal{F} = \mathbf{V}(q_1, \ldots, q_\omega) \subset \mathbb{R}^{2N} \tag{13}$$

identifies the desired controlled formation of the agents. Essentially, the former set determines the absolute desired patrolling paths to be assigned to some or to all the agents in the team, whereas the latter set characterizes the potential relative displacement of some of the robots (*followers*, if any), with respect to others (*leaders*, if any). Consider the following proposition, whose proof is given in Appendix B.

Proposition 1. *The output of Algorithm 1 with input*

$$p_{1,1}, \ldots, p_{1,s_1}, p_{2,1}, \ldots, p_{2,s_2}, \ldots, p_{N,s_N}, q_1, \ldots, q_\omega \in \mathbb{R}[x],$$

is a family of vector fields $h(x)$ such that the corresponding dynamical system $\dot{x} = h(x)$ has the affine variety

$$\mathcal{V} = \mathcal{F} \cap \left(\bigcap_{i=1}^{N} \mathcal{V}_i \right) \subset \mathbb{R}^{2N}. \tag{14}$$

as attractive in $\mathcal{I}_\mathcal{V}$ and h-invariant set.

Note that if the affine variety $\mathcal{V}$ given in (14) is empty, then, by construction, Algorithm 1 returns $\mathcal{I}_\mathcal{V} = \emptyset$, thus showing unfeasibility of the considered problem. Interestingly, despite the fact that the statement of Proposition 1 merely guarantees the existence of the vector field h, indeed the results of [36] allow also to explicitly characterize, and especially *parameterize*, the structure of such vector fields. More precisely, it has been shown in [36] that the set of $h \in \mathbb{R}^{2N}[x]$ provided as outputs of Algorithm 1 with input $p_{1,1}, \ldots, p_{1,s_1}$, $p_{2,1}, \ldots, p_{2,s_2}, \ldots, p_{N,s_N}, q_1, \ldots, q_\omega$ is described by

$$h(x) = g(x)\chi + f_b(x), \tag{15}$$

where

$$\left[\begin{array}{c|c} g & f_b \\ \hline 0 & 1 \end{array}\right] = \mathrm{Syz}([\ \nabla p \quad \Lambda p\ ]),$$

where $p = [\ p_{1,1} \quad \ldots \quad q_\omega\ ]^\top$ and χ is an arbitrary vector, potentially function of the state variable, which might be employed to induce additional motions or

to optimize desired optimality criteria, without hindering the achievement of the primary task of patrolling and formation control.

The results of Proposition 1 permit the design of patrolling paths for the entire *multi-agent* model, while also potentially establishing hierarchical ordering or formations among the agents. Namely, by [37], the affine variety $\mathcal{V}$ is invariant and locally attractive with respect to the system

$$\dot{\xi}(t) = g(\xi(t))\chi + f(\xi(t)), \tag{16}$$

with basin of attraction containing the set $\mathcal{I}_{\mathcal{V}}$. However, such techniques do not guarantee that collisions during motions are avoided. This is the objective of the following section.

3.3. Collision avoidance among agents and with obstacles

The main objective of this section consists in designing a vector field $\sigma(x)$ - to be combined with the family yielded by Proposition 1 as pursued in the subsequent Section 3.4 - such that the trajectories of the dynamical system $\dot{x} = \sigma(x)$ do not collide with the obstacles. Toward this end, the definitions of obstacle and collision are stated below.

Definition 2. *(Obstacle).* The region in the configuration space occupied by the w-th obstacle is denoted $\mathcal{O}_w$. There exists a polynomial $\varsigma_w \in \mathbb{R}[x_1, x_2]$ such that the boundary $\partial\mathcal{O}_w$ of $\mathcal{O}_w$ is given by

$$\partial\mathcal{O}_w = \mathbf{V}(\varsigma_w). \tag{17}$$

In this framework, the size of the agents is encoded in the polynomials ς_w, with $w = 1, \ldots, W$, that are suitably enlarged in order to take into account also the volume of each agent. It is worth noticing that, as it is customary in this context, it is possible to enclose obstacles whose boundary is not polynomial within an associated ellipse of minimal size.

Definition 3. *(Collision).* Suppose there are W obstacles described by the regions $\mathcal{O}_w$, $w = 1, \ldots, W$. Let r_i characterize the size of the i–th agent,

$i = 1, \ldots, N$. A collision is said to occur if there exists $t \in \mathbb{R}_{\geqslant 0}$ such that either $x_i(t) \in \partial\mathcal{O}_w(x_i)$ for some $i \in \{1, \ldots, N\}$, or $|x_i(t) - x_j(t)| = r_i + r_j$ for some $i, j \in \{1, \ldots, N\}$, $i \neq j$.

Define, for $i, j \in \{1, \ldots, N\}$, $i \neq j$, the polynomial

$$c_{i,j} = (x_{i,1} - x_{j,1})^2 + (x_{i,2} - x_{j,2})^2 - (r_i + r_j)^2 \tag{18}$$

in $\mathbb{R}[x]$, and let

$$\mathcal{P}_{i,j} = \mathbf{V}(c_{i,j}) \subset \mathbb{R}^{2N}.$$

Note that $x \in \mathcal{P}_{i,j}$ if and only if $|x_i(t) - x_j(t)| = r_i + r_j$.

In the following, we suppose that the desired patrolling path is *feasible* in terms of obstacles and prescribed formations, as formalized in the following statement.

Assumption 3. The patrolling path, controlled formation and obstacles are such that

$$\mathcal{V} \cap \left(\bigcup_{w=1}^{W} \mathcal{O}_w \right) \cap \left(\bigcup_{i=1}^{N} \bigcup_{j=i+1}^{N} \mathcal{P}_{i,j} \right) = \emptyset,$$

where $\mathcal{V}$ is defined as in (14).

In order to ensure that Assumption 3 is met and to employ the collision avoidance strategy described in the remainder of this section, the shapes of the obstacles have to be known in advance. In the following lemma (whose proof is given in Appendix C), we propose a tool to assess whether a collision has occurred.

Lemma 1. *Define the polynomial in* $\mathbb{R}[x]$

$$b(x) = \left(\prod_{i=1}^{N} \prod_{w=1}^{W} \varsigma_w(x_i) \right) \cdot \left(\prod_{i=1}^{N} \prod_{j=i+1}^{N} c_{i,j}(x) \right).$$

A collision occurs if and only if there is $t \in \mathbb{R}_{\geqslant 0}$ *such that*

$$b(x(t)) = 0.$$

Assumption 3 does not depend on the initial configuration of the agents and it may be equivalently - but potentially less intuitive from the geometric point of view - stated by requiring that $\mathcal{V}$ defined as in (14) and $\mathbf{V}(b)$ possess an empty intersection, *i.e.* $\mathcal{V} \cap \mathbf{V}(b) = \emptyset$. This latter condition is, on the other hand, more prone than Assumption 3 to be systematically checked. By taking advantage of the result given in Lemma 1 and by considering a suitable adaptation of classical barrier functions, in the following we design vector fields $\sigma(x)$ such that the trajectories of the corresponding dynamical system $\dot{x} = \sigma(x)$ do not collide with the obstacles. Toward this end, let $p_{1,1}, \ldots, p_{1,s_1}, p_{2,1}, \ldots, p_{2,s_2}, \ldots, p_{N,s_N}, q_1, \ldots, q_\omega \in \mathbb{R}[x]$ be defined as in Section 3.2 and let

$$a = p_{1,1}^2 + \cdots + p_{1,s_1}^2 + \cdots + p_{N,s_N}^2 + q_1^2 + \cdots + q_\omega^2.$$

Thus, letting b be defined as in Lemma 1, consider

$$r(x) = \frac{a(x)}{b(x)}. \tag{19}$$

By relying on Assumption 3, the function r goes to zero as x tends to the affine variety $\mathcal{V}$ given in (14) and tends to infinity as $b(x)$ tends to 0 (*i.e.*, if a collision is about to occur). Therefore, the positive invariance of sub-level sets of the function $r(x)$ with respect to the system $\dot{x} = \sigma(x)$ implies that the trajectories of such a system generate collision-free path for the multi-agent system. Hence, let

$$\eta(x) = \nabla r(x), \tag{20}$$

and consider the following (rational) system

$$\dot{x}(t) = -\eta(x(t))\beta(t), \tag{21}$$

where $\beta : \mathbb{R}_{\geqslant 0} \to \mathbb{R}_{\geqslant 0}$. The following proposition (whose proof is given in Appendix D) shows that the trajectories of system (21) are collision-free.

Proposition 2. *Let $x_0 \in \mathbb{R}^{2N}$ be given and assume that $r(x_0) \geqslant 0$. Suppose that Assumption 3 holds. If $r(x_0) < \infty$, then, letting $x(t)$ be the solution of system* (21)*, there does not exist a time $t \geqslant 0$ such that $b(x(t)) = 0$.*

Hence, by Proposition 2 and Lemma 1, the vector field $\sigma(x) = -\eta(x)\beta$ solves the collision avoidance problem. It is worth noticing that the main goal of such a vector field is just to ensure that the path of motion of the mobile robots, modeled by system (21), is collision free. Such a goal can be pursued using an unified framework to deal with collision both among agents and with obstacles since the design is centralized, *i.e.*, the motion of all the agents is jointly designed toward this objective. Such a centralization avoids the issues that may arise from the disjoint design of the motion of all the mobile robots, which should account for the fact that the motion of each agent is influenced and influences the one of all the others.

3.4. Patrolling, formation control and collision avoidance

The control tools proposed in Section 3.2 constitute a solution to the patrolling and formation control goals alone, whereas the method given in Section 3.3 only guarantees that the trajectories of the multi-agent system are collision-free. The main objective of this section is to show how to suitably combine these two results together to solve the patrolling in formation and collision avoidance problem. Toward this end, consider the following assumption.

Assumption 4. Let $\mathcal{V}$ be defined as in (14) and let an initial configuration $x_0 \in \mathbb{R}^{2N}$ be given. There exists a continuous path $\mathcal{P} \subset \mathbb{R}^{2N}$ between x_0 and $\mathcal{V}$ such that $\mathcal{P} \cap \mathbf{V}(b) = \emptyset$.

Assumption 4 essentially guarantees that there exists a solution to the patrolling in formation and collision avoidance problem for a given initial configuration of the agents. The control task basically consists in determining such feasible path $\mathcal{P}$. Thus, let $p_{1,1}, \ldots, p_{1,s_1}, p_{2,1}, \ldots, p_{2,s_2}, \ldots, p_{N,s_N}, q_1, \ldots, q_\omega \in \mathbb{R}[x]$ be defined as in Section 3.2, let h (parametrized as in (15), with $f_b \in \mathbb{R}^{2N}[x]$ and $g \in \mathbb{R}^{2N\times\hbar}[x]$) be the output of Algorithm 1, and let η and b be defined as in Section 3.3. By Propositions 1 and 2 and by Lemma 1, the most intuitive way to combine the two previously designed vector fields would be to exploit the degrees of freedom hinted at in the discussion after Proposition 1 to *shape*

the vector field as the one in (21) for some β. Hence, if there exist functions $\bar{\chi} : \mathbb{R}^{2N} \to \mathbb{R}^{\hbar}$ and $\bar{\beta} : \mathbb{R}^{2N} \to \mathbb{R}_{\geqslant 0}$ such that $f_b(x) + g(x)\bar{\chi}(x) = \eta(x)\bar{\beta}(x)$, then the solutions to the dynamical system

$$\dot{x} = f_b(x) + g(x)\bar{\chi}(x)$$

would solve the patrolling in formation and collision avoidance problems simultaneously. However, such functions need not exist in general, and alternative *combination* strategies must be envisioned. Thus, consider the following system

$$\dot{x} = \gamma_1 b^2(x) f_b(x) + \gamma_1 b^2(x) g(x) \zeta(x) - \eta(x)\mu(x)p(x), \tag{22}$$

where $p(x) := [\ p_{1,1} \quad \cdots \quad p_{N,s_N} \quad q_1 \quad \ldots \quad q_\omega\ ]^\top \in \mathbb{R}^{\varkappa}[x]$ and $\gamma_1 > 0$, which has been obtained by adding the dynamics of system (16) multiplied by $\gamma_1 b^2(x)$ and (21), and by letting $\chi = \zeta(x)$ and $\beta = \mu(x)p(x)$, where $\zeta : \mathbb{R}^{2N} \to \mathbb{R}^{\hbar}$ and $\mu : \mathbb{R}^{2N} \to \mathbb{R}^{1\times\varkappa}$. The following theorem (whose proof is given in Appendix E) states that, if the function μ and the parameter γ_1 in (22) satisfy some easily verifiable assumptions, then system (22) locally solves the patrolling in formation and collision avoidance problem.

Theorem 2. *Let Assumptions 3 and 4 hold and let the position and the shape of the obstacles $\mathcal{O}_w$, $w = 1, \ldots, W$, be known. Thus, let $\mathcal{R} := (\bigcup_{i=1}^{N}\bigcup_{w=1}^{W}\{x \in \mathbb{R}^{2N} : \varsigma_w(x_i) \leqslant 0\}) \cdot (\bigcup_{i=1}^{N}\bigcup_{j=i+1}^{N}\{x \in \mathbb{R}^{2N} : c_{i,j}(x) \leqslant 0\})$ be the set of unfeasible states and let Λ be the matrix defined at Step 3 of Algorithm 1. Thus, if $\mu : \mathbb{R}^{2N} \to \mathbb{R}^{1\times\omega}$ and γ_1 are such that:*

(a) the symmetric part of the matrix $\gamma_1 b^2 \Lambda + \frac{\partial p}{\partial x}\eta\mu$ is positive definite in $\mathcal{Y} := \{x \in \mathbb{R}^{2N} \setminus \mathcal{R} : p^\top(x)p(x) \leqslant d\}$, for some sufficiently small $d \in \mathbb{R}_{\geqslant 0}$;

(b) $\mu(x)p(x) \geqslant 0$ for all $x \in \mathbb{R}^{2N} \setminus \mathcal{R}$;

then system (22) solves the patrolling in formation and collision avoidance problem for all $x_0 \in \mathcal{Y}$.

Note that the choice $\gamma_1 > 0$ and $\mu(x) = \gamma_2 p^\top(x)$, with $\gamma_2 > 0$, guarantees that the requirements of Theorem 2 are satisfied for a sufficiently small $d \in \mathbb{R}_{>0}$

under Assumption 3. As a matter of fact, since, by Assumption 3, one has $\mathcal{V} \cap \mathcal{R} = \emptyset$, there exists a sufficiently small $\varepsilon^\star \in \mathbb{R}_{>0}$ such that $b(x) \neq 0$ for all $x \in \mathcal{V} + \varepsilon^\star \mathcal{B}$. Therefore, one has that $\gamma_1 b^2 \Lambda + \gamma_2 \frac{\partial p}{\partial x} \eta p^\top \simeq \gamma_1 b^2 \Lambda$ for all $x \in \mathcal{V} + \varepsilon^\star \mathcal{B}$, thus implying that in $\mathcal{V} + \varepsilon^\star \mathcal{B}$ the dynamics of system (22) essentially matches the one of system (16) rescaled via the constant γ_1. Clearly, the domain $\mathcal{V} + \varepsilon^\star \mathcal{B}$ is a restrictive estimate of the set of initial conditions for which system (22) solves the patrolling in formation and collision avoidance problem. In particular, thanks to the use of the hybrid implementation (see the subsequent Theorem 3), the basin of attraction of the desired formation is usually almost all the workspace of the multi-agent system; see [30] for further details.

Therefore, letting $k = \mathrm{col}(\gamma_1, \gamma_2, \chi)$ and defining the set of admissible parameters $\mathcal{A}$ as any compact subset of $\mathbb{R}_{>0} \times \mathbb{R}_{>0} \times \mathbb{R}^s$, the parametric vector field

$$f(x, k) = \gamma_1 b^2(x) f_b(x) + \gamma_1 b^2(x) g(x) \chi - \gamma_2 \|p(x)\|_2^2 \eta(x), \tag{23}$$

locally solves the problem of designing a path for mobile robots that is collision free and that steers them to patrol a preassigned curve in controlled formation. Furthermore, letting

$$\underline{\gamma}_1 := \min_{\mathcal{A}} \gamma_1, \qquad\qquad \underline{b} := \min_{x \in \mathcal{A} + \varepsilon^\star \mathcal{B}} b(x),$$

which are both positive under Assumption 4 since $\mathcal{V} \cap \mathbf{V}(b) = \emptyset$, the vector field $f(x, k)$, together with the functions

$$V(x) = \|p(x)\|_2^2, \qquad\qquad \varrho(x) = \frac{1}{2} \underline{\gamma}_1 \, \underline{b} \, p^\top \Lambda p, \tag{24}$$

satisfies Assumptions 1 and 2, thus allowing us to implement system (22) via the hybrid implementation (10).

Therefore, the hybrid implementation (10), with f defined as in (23), locally solves the patrolling in formation and collision avoidance problem, as formally stated in the following theorem, whose proof is reported in Appendix F.

Theorem 3. *Suppose that Assumptions 3 and 4 hold. Let f be defined as in* (23)*, let $\mathcal{A}$ be a compact subset of $\mathbb{R}_{>0} \times \mathbb{R}_{>0} \times \mathbb{R}^s$, and consider the hybrid*

system (10). *If the function* $\Pi(x, k, \kappa)$ *is level bounded in* κ *locally uniformly in* (x, k) *and the function* $\varpi(x, k) := \inf_{\kappa \in \Xi(x,k)} \Pi(x, k, \kappa)$ *is locally bounded from above and* C^0*, then the hybrid implementation* (10) *solves the patrolling in formation and collision avoidance problem for all* $x_0 \in \mathcal{Y}$.

By Theorem 3, if the inputs of (3) are selected as

$$(x, k) \in \mathcal{C}, \quad u = f(x, k), \tag{25a}$$

$$(x, k) \in \mathcal{D}, \quad k^+ \in \operatorname*{argmin}_{\kappa \in \Xi(x,k)} \Pi(x, k, \kappa), \tag{25b}$$

then the closed loop trajectories constitute paths that are collision free and that steer the robots to patrol a preassigned path in controlled formation.

In view of the discussion given in Section 1.2, such a centralized control strategy can be implemented by the team of mobile robots by letting their inputs be

$$\begin{bmatrix} v_1 \\ \omega_1 \\ \vdots \\ v_N \\ \omega_N \end{bmatrix} = \begin{bmatrix} \cos(\Theta_1) & \sin(\Theta_1) & \cdots & 0 & 0 \\ -\dfrac{\sin(\Theta_1)}{L_1} & \dfrac{\cos(\Theta_1)}{L_1} & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & \cos(\Theta_N) & \sin(\Theta_N) \\ 0 & 0 & \cdots & -\dfrac{\sin(\Theta_N)}{L_N} & \dfrac{\cos(\Theta_N)}{L_N} \end{bmatrix} f(x, k), \tag{26}$$

where $x = \begin{bmatrix} X_{P,1} & Y_{P,1} & \cdots & X_{P,N} & Y_{P,N} \end{bmatrix}^\top$ and the navigation parameters k are updated following the hybrid dynamics (25).

4. Simulation results

In this section, the techniques outlined in Sections 2 and 3 are tested via numerical simulations. In particular, Example 1 shows how the proposed hybrid mechanism can be used to steer the agents to desired target positions, Example 2 shows how it can be used to let the agents patrol a selected algebraic curve, and Example 3 shows how it can be used to let the leader patrol a selected curve while the other agents keep a prescribed formation.

Example 1. Let $N = 4$ and suppose that the size of the 4 agents is $r_i = 0.3$, $i = 1, \ldots, 4$. The objective of this example is to steer the four robots to the target points

$$x_{t,1} = \begin{bmatrix} -1 \\ -0.5 \end{bmatrix}, \qquad x_{t,2} = \begin{bmatrix} 1 \\ 0.5 \end{bmatrix},$$
$$x_{t,3} = \begin{bmatrix} -1 \\ 0.5 \end{bmatrix}, \qquad x_{t,4} = \begin{bmatrix} 1 \\ -0.5 \end{bmatrix},$$

Thus, define the polynomials

$$p_{1,1} = x_{1,1} + 1, \qquad p_{1,2} = x_{1,2} + 0.5,$$
$$p_{2,1} = x_{2,1} - 1, \qquad p_{2,2} = x_{2,2} - 0.5,$$
$$p_{3,1} = x_{3,1} + 1, \qquad p_{3,2} = x_{3,2} - 0.5,$$
$$p_{4,1} = x_{4,1} - 1, \qquad p_{4,2} = x_{4,2} + 0.5.$$

Algorithm 1 with input $p_{1,1}, \ldots, p_{4,2}$ returns a vector field h, parametrized as in (15) with $f_b \in \mathbb{R}^8[x]$ and $g = 0$,

$$f_b(x) = \begin{bmatrix} -x_{1,1}-1 \\ -x_{1,2}-0.5 \\ 1-x_{2,1} \\ 0.5-x_{2,2} \\ -x_{3,1}-1 \\ 0.5-x_{3,2} \\ 1-x_{4,1} \\ -x_{4,2}-0.5 \end{bmatrix},$$

and the matrix $\Lambda = I_8$.

Assume that, in the workspace of the multi-agent system, there are four obstacles $\mathcal{O}_w$, whose boundary is $\partial\mathcal{O}_w = \mathbf{V}(\varsigma_w) \subset \mathbb{R}^2$, $\varsigma_w \in \mathbb{R}[x_1, x_2]$, $w = 1, \ldots, 4$, where

$$\varsigma_1 = x_1 - 1.4, \qquad \varsigma_2 = x_1 + 1.4, \tag{27a}$$
$$\varsigma_3 = x_2 + 0.8, \qquad \varsigma_4 = x_2 - 0.8, \tag{27b}$$

so that the actual workspace of the multi-agent system is the interior of the rectangle $[-1.4, 1.4] \times [-0.8, 0.8] \subset \mathbb{R}^2$.

Following Proposition 2 and Lemma 1, let

$$a = \sum_{i=1}^{N}\sum_{j=1}^{2} p_{i,j}^2 \in \mathbb{R}[x],$$
$$b = \left(\prod_{w=1}^{4} \varsigma_w(x_1)\varsigma_w(x_2)\varsigma_w(x_3)\varsigma_w(x_4)\right)\left(\prod_{i=1}^{4}\prod_{j=i+1}^{4} c_{i,j}(x)\right) \in \mathbb{R}[x],$$

where the polynomials $c_{i,j}$ are as in (18), let $r(x) = \frac{p(x)}{b(x)}$, and let $\eta(x) = \frac{\partial r(x)}{\partial x}$. Hence, defining $f(x,k)$ as in (23), the trajectories of the hybrid implementation (10) of system (5) are collision-free path of motions for the multi-agent system.

Figure 3 depicts the solution to system (10) with $\mu = 10^{-2}$, $x(0) = [\ -1.2 \quad 0 \quad -0.4 \quad 0 \quad 0.4 \quad 0 \quad 1.2 \quad 0\ ]^\top$, $\mathcal{A} = [10^{-3}, 10^3]^2$, $\Pi(k,x,\kappa) = \kappa_1^2 + \kappa_2^2$, and $k(0) = [\ 1 \quad 0.5\ ]^\top$.

As shown by such a figure, the hybrid implementation (10) generates collision-free path of motions for the mobile robots that avoid collision and that steers the agents toward the desired target points. It is worth noticing that the hybrid implementation triggers a change in the parameters k at time $t = 9.4818 \cdot 10^{-4}$ so to guarantee convergence of the path of motion to the desired target points. This emphasizes the fact that the proposed navigation algorithm is capable of autonomously changing the initial values of the parameters k so to ensure convergence. $\triangle$

Example 2. Let $N = 3$ and suppose that the size of the 3 agents is $r_i = 0.3$, $i = 1, 2, 3$. The objective of this example is to let the three robots patrol the affine variety (see Figure 4)

$$\mathbf{V}\left(x_1^4 - x_1^2 + x_2^2 - \tfrac{1}{100}\right)$$

Thus, define the polynomials

$$p_{1,1} = x_{1,1}^4 - x_{1,1}^2 + x_{1,2}^2 - \tfrac{1}{100},$$
$$p_{2,1} = x_{2,1}^4 - x_{2,1}^2 + x_{2,2}^2 - \tfrac{1}{100},$$
$$p_{3,1} = x_{3,1}^4 - x_{3,1}^2 + x_{3,2}^2 - \tfrac{1}{100}.$$

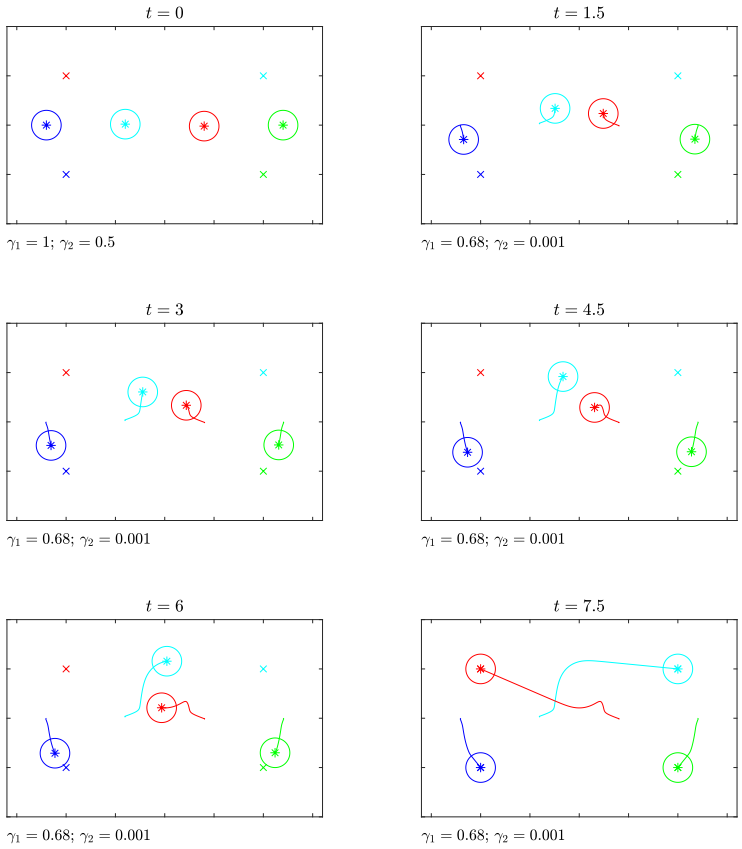


Figure 3: Solution of the hybrid implementation (10) for the problem considered in Example 1.

Algorithm 1 with input $p_{1,1}, p_{2,1}, p_{3,1}$ returns a vector field h, parametrized as in (15) with $f_b \in \mathbb{R}^6[x]$ and $g \in \mathbb{R}^{6\times 3}[x]$,

$$f_b(x) = \begin{bmatrix} -x_{1,1}^7 + \frac{3x_{1,1}^5}{2} - x_{1,2}^2 x_{1,1}^3 - \frac{49x_{1,1}^3}{100} + \frac{1}{4}x_{1,2}^2 x_{1,1} - \frac{x_{1,1}}{200} \\ -\frac{1}{2}x_{1,2}^3 + \frac{1}{4}x_{1,1}^2 x_{1,2} + \frac{x_{1,2}}{200} \\ -x_{2,1}^7 + \frac{3x_{2,1}^5}{2} - x_{2,2}^2 x_{2,1}^3 - \frac{49x_{2,1}^3}{100} + \frac{1}{4}x_{2,2}^2 x_{2,1} - \frac{x_{2,1}}{200} \\ -\frac{1}{2}x_{2,2}^3 + \frac{1}{4}x_{2,1}^2 x_{2,2} + \frac{x_{2,2}}{200} \\ -x_{3,1}^7 + \frac{3x_{3,1}^5}{2} - x_{3,2}^2 x_{3,1}^3 - \frac{49x_{3,1}^3}{100} + \frac{1}{4}x_{3,2}^2 x_{3,1} - \frac{x_{3,1}}{200} \\ -\frac{1}{2}x_{3,2}^3 + \frac{1}{4}x_{3,1}^2 x_{3,2} + \frac{x_{3,2}}{200} \end{bmatrix},$$

$$g(x) = \begin{bmatrix} 0 & 0x_{1,2} \\ 0 & 0x_{1,1} - 2x_{1,1}^3 \\ 0 & x_{2,2} \quad 0 \\ 0 & x_{2,1} - 2x_{2,1}^3 \quad 0 \\ x_{3,2} & 0 \quad 0 \\ x_{3,1} - 2x_{3,1}^3 & 0 \quad 0 \end{bmatrix},$$

and the matrix

$$\Lambda = \operatorname{diag}\Big(x_{1,1}^2(1-2x_{1,1}^2)^2 + x_{1,2}^2, \\ x_{2,1}^2(1-2x_{2,1}^2)^2 + x_{2,2}^2, x_{3,1}^2(1-2x_{3,1}^2)^2 + x_{3,2}^2 \Big).$$

Letting the workspace be as in Example 1, define the polynomials $\varsigma_1, \ldots, \varsigma_4$ as in (27), let

$$a = \sum_{i=1}^{N} p_{i,1}^2 \in \mathbb{R}[x],$$
$$b = \left(\prod_{w=1}^{4} \varsigma_w(x_1)\varsigma_w(x_2)\varsigma_w(x_3) \right) \left(\prod_{i=1}^{3} \prod_{j=i+1}^{3} c_{i,j}(x) \right) \in \mathbb{R}[x],$$

where the polynomials $c_{i,j}$ are as in (18), let $r(x) = \frac{p(x)}{b(x)}$, and let $\eta(x) = \frac{\partial r(x)}{\partial x}$. Hence, letting $f(x,k)$ be defined as in (23), the trajectories of the hybrid implementation (10) of system (5) are collision-free path of motions for the multi-agent system.

Figure 4 depicts the solution to system (10) with $\mu = 10^{-2}$, $x(0) = [\ -0.5 \quad 0.5 \quad 0 \quad 0.5 \quad 0.5 \quad 0.5\ ]^\top$, $\mathcal{A} = [10^3, 10^6] \times [10^{-3}, 10^3]$, $\Pi(k, x, \kappa) = \kappa_1^2 + \kappa_2^2 + (\kappa_3 - 1)^2 + (\kappa_4 - 1)^2 + (\kappa_5 - 1)^2$ (where κ_1 and κ_2 have the role of γ_1 and γ_2, respectively, whereas $[\ \kappa_3 \quad \kappa_4 \quad \kappa_5\ ]^\top$ have the role of χ), and $k(0) = [\ 10^3 \quad 10^{-2} \quad 1 \quad 1 \quad 1\ ]^\top$.

As shown by such a figure, the hybrid implementation (10) generates collision-free path of motions for the mobile robots that avoids collision and that let the agents patrol the desired curve. Note that the trajectories attained by the agents are dependent just on the initial condition $(x(0), k(0))$, which uniquely determines the state-evolution of the (autonomous) hybrid system (10). Note that the hybrid implementation triggers a change in the parameters k at time $t = 1.4950$ so to guarantee convergence of the path of motion to $\mathcal{V}$. $\triangle$

Example 3. Let $N = 3$ and suppose that the size of the 3 agents is $r_i = 0.3$, $i = 1, 2, 3$. The objective of this example is to let the first robot (the leader)

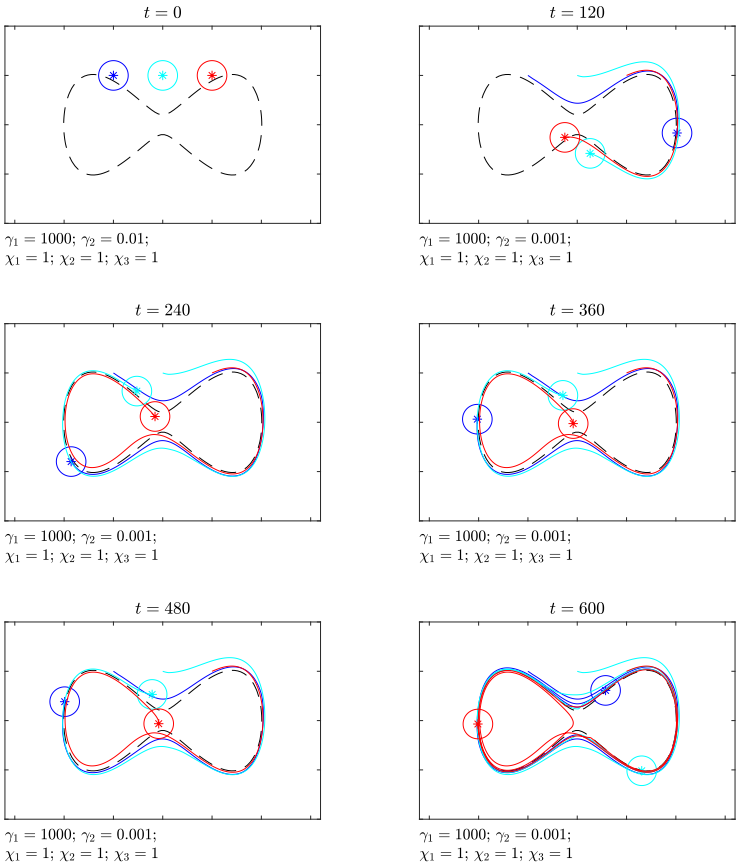


Figure 4: Solution of the hybrid implementation (10) for the problem considered in Example 2.

patrol the circle

$$\mathbf{V}\left(x_1^2 + x_2^2 - \tfrac{1}{8}\right),$$

while letting the other two agents be at prescribed distance, equal to 0.5. Thus, define the polynomials

$$\begin{aligned} p_{1,1} &= x_{1,1}^2 + x_{1,2}^2 - \tfrac{1}{8}, \\ q_1 &= (x_{1,1} - x_{2,1})^2 + (x_{1,2} - x_{2,2})^2 - \tfrac{1}{4}, \\ q_2 &= (x_{1,1} - x_{3,1})^2 + (x_{1,2} - x_{3,2})^2 - \tfrac{1}{4}. \end{aligned}$$

Algorithm 1 with input $p_{1,1}, q_1, q_2$ returns a vector field h, parametrized as in

(15) with $f_b \in \mathbb{R}^6[x]$ and $g \in \mathbb{R}^{6\times 3}[x]$,

$$f_b(x) = \begin{bmatrix} -\frac{1}{2}x_{1,1}^3 - x_{1,2}^2 x_{1,1} + \frac{x_{1,1}}{16} \\ \frac{x_{1,2}}{16} - \frac{x_{1,2}^3}{2} \\ f_{b,3}(x) \\ -\frac{1}{2}x_{2,2}^3 + \frac{3}{2}x_{1,2}x_{2,2}^2 - \frac{3}{2}x_{1,2}^2 x_{2,2} + \frac{x_{2,2}}{8} - \frac{x_{1,2}}{16} \\ f_{b,5}(x) \\ -\frac{1}{2}x_{3,2}^3 + \frac{3}{2}x_{1,2}x_{3,2}^2 - \frac{3}{2}x_{1,2}^2 x_{3,2} + \frac{x_{3,2}}{8} - \frac{x_{1,2}}{16} \end{bmatrix},$$

$$g(x) = \begin{bmatrix} 0 & 0 & x_{1,2} \\ 0 & 0 & -x_{1,1} \\ 0 & x_{1,2}-x_{2,2} & x_{2,2} \\ 0 & x_{2,1}-x_{1,1} & -x_{2,1} \\ x_{1,2}-x_{3,2} & 0 & x_{3,2} \\ x_{3,1}-x_{1,1} & 0 & -x_{3,1} \end{bmatrix},$$

where

$$\begin{aligned} f_{b,3}(x) = &-\tfrac{1}{2}x_{2,1}^3 + \tfrac{3}{2}x_{1,1}x_{2,1}^2 - \tfrac{3}{2}x_{1,1}^2 x_{2,1} - x_{1,2}^2 x_{2,1} \\ &- x_{2,2}^2 x_{2,1} + 2x_{1,2}x_{2,2}x_{2,1} \\ &+ \tfrac{1}{8}x_{2,1} + x_{1,1}x_{2,2}^2 - \tfrac{1}{16}x_{1,1} - 2x_{1,1}x_{1,2}x_{2,2}, \\ f_{b,5}(x) = &-\tfrac{1}{2}x_{3,1}^3 + \tfrac{3}{2}x_{1,1}x_{3,1}^2 - \tfrac{3}{2}x_{1,1}^2 x_{3,1} - x_{1,2}^2 x_{3,1} \\ &- x_{3,2}^2 x_{3,1} + 2x_{1,2}x_{3,2}x_{3,1} \\ &+ \tfrac{1}{8}x_{3,1} + x_{1,1}x_{3,2}^2 - \tfrac{1}{16}x_{1,1} - 2x_{1,1}x_{1,2}x_{3,2}, \end{aligned}$$

and the matrix

$$\Lambda = \operatorname{diag}\Big(x_{1,1}^2 + x_{1,2}^2, (x_{1,1}-x_{2,1})^2 + (x_{1,2}-x_{2,2})^2, \\ (x_{1,1}-x_{3,1})^2 + (x_{1,2}-x_{3,2})^2 \Big).$$

Letting the workspace be as in Examples 1 and 2, define the vector η as in Example 2. Hence, letting $f(x,k)$ be defined as in (23), by Theorem 3, the trajectories of the hybrid implementation (10) of system (5) are collision-free path of motions for the multi-agent system.

Figure 5 depicts the solution to system (10) with $\mu = 10^{-2}$, $x(0) = [\ -1 \quad -0.6 \quad -0.5 \quad 0 \quad -1 \quad 0.6\ ]^\top$, $\mathcal{A} = [10^{-3}, 10^3]^2$, $\Pi(k,x,\kappa) = \kappa_1^2 + \kappa_2^2 + \kappa_3^2 + \kappa_4^2 + (\kappa_5 - 1)^2$ (where κ_1 and κ_2 have the role of γ_1 and γ_2, respectively, whereas $[\ \kappa_3 \quad \kappa_4 \quad \kappa_5\ ]^\top$ have the role of χ), and $k(0) = [\ 10^3 \quad 10^{-3} \quad 0 \quad 0 \quad 1\ ]^\top$.

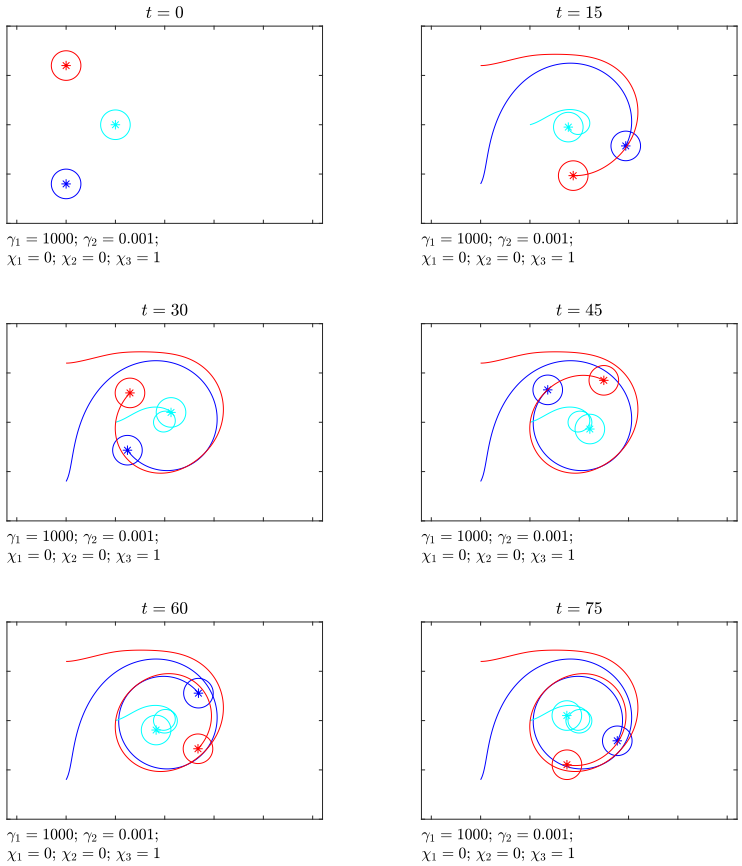


Figure 5: Solution of the hybrid implementation (10) for the problem considered in Example 3.

As shown by such a figure, the hybrid implementation (10) generates collision-free path of motions for the mobile robots that avoid collision and that let the leader patrol the desired curve while letting the other agents be in formation. It is worth noticing that, in such a simulation, the hybrid implementation (10) does not trigger any change in the constants k. △

5. Experimental results

Experiments have been carried out to test the control law derived in Examples 1, 2, and 3 using the `Robotarium`, a remotely accessible robotic testbed, designed to help users quickly prototype and validate their control strategies

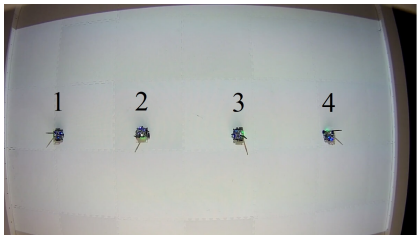
through implementation on physical robots without the overhead of setting up their own hardware or high-fidelity simulation. In particular, Experiment 1 shows how the control law given in Example 1 can be employed to steer four mobile robots to a desired target point. In Experiment 2, the control method outlined in Example 2 is used to let three mobile agents patrol a given curve. In the last experiment, the control law reported in Example 3 is validated, letting the leader patrol a circle while keeping the other two agents at a prescribed distance. To ease the description of the experimental results, a set of relevant snapshots is presented, supported by a proper labeling of the involved robots.

The single-integrator dynamics corresponding to the hybrid implementation (10) have been converted to unicycle dynamics using the linearizing feedback given in (2), with $L = 0.1\,\mathrm{m}$, *i.e.*, the team of mobile robots has been controlled using the centralized control law given in (26). Further, since the trajectories of the system $\dot{x}(t) = \digamma(t) f(x(t))$ are the same as those of the system $\dot{x}(t) = f(x(t))$, provided that $\digamma(t) > 0$ for all $t \in \mathbb{R}$, the linear and angular velocities resulting from (2) have been uniformly scaled so to meet the actuation constraint. Namely, letting v_i and ω_i be the linear and angular velocities resulting from (2) for$i = 1, \ldots, N$, and letting $\bar{v}$ and $\bar{\omega}$ be the maximal admissible linear and angular velocities of the mobile robots, the parameter $\digamma(t)$ above has been selected as

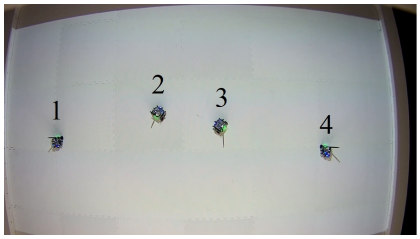
$$\digamma = \min\left\{1,\ \frac{\bar{v}}{2\max_i\{|v_i|\}},\ \frac{\bar{\omega}}{2\max_i\{|\omega_i|\}}\right\}.$$

Experiment 1. The control law given in Example 1 has been used to steer four robots to a target position. The initial poses of the robots are $X_1(0) = -1.2\,\mathrm{m}$, $Y_1(0) = 0\,\mathrm{m}$, $\Theta_1(0) = 0\,\mathrm{rad}$, $X_2(0) = -0.4\,\mathrm{m}$, $Y_2(0) = 0\,\mathrm{m}$, $\Theta_2(0) = \frac{\pi}{2}\,\mathrm{rad}$, $X_3(0) = 0.4\,\mathrm{m}$, $Y_3(0) = 0\,\mathrm{m}$, $\Theta_3(0) = \pi\,\mathrm{rad}$, $X_4(0) = 1.2\,\mathrm{m}$, $Y_4(0) = 0\,\mathrm{m}$, $\Theta_4(0) = \frac{3}{2}\pi\,\mathrm{rad}$.

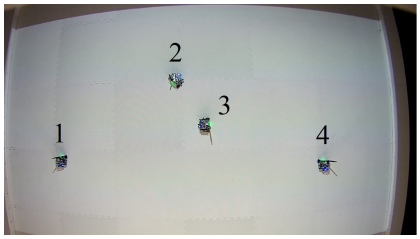
Figure 6 reports some snapshots of the corresponding experiment; see [38] for the full video. As shown by such a figure, the proposed control method allows to steer the four agents to their target positions avoiding collisions among themselves and with fixed obstacles despite the actuators saturation and the



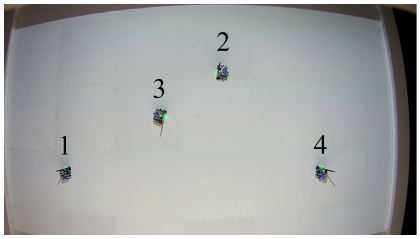
(a) Time $t = 0\,\mathrm{s}$.



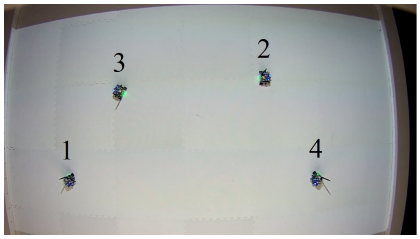
(b) Time $t = 4\,\mathrm{s}$.



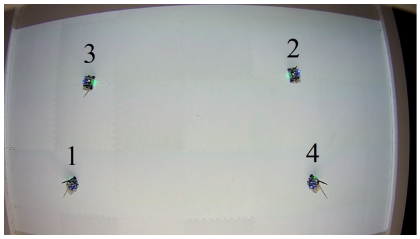
(c) Time $t = 8\,\mathrm{s}$.



(d) Time $t = 12\,\mathrm{s}$.



(e) Time $t = 16\,\mathrm{s}$.



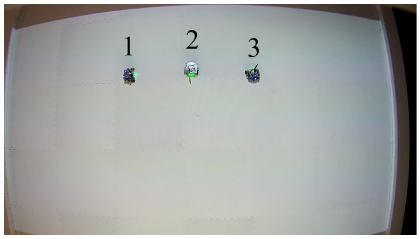
(f) Time $t = 20\,\mathrm{s}$.

Figure 6: Result of the experiment with the control law given in Example 1.

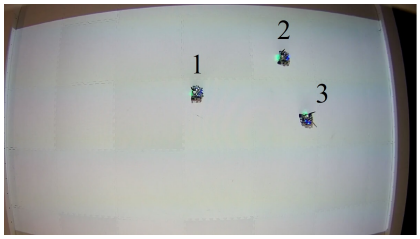
robots nonidealities. However, by comparing Figures 3 and 6, it can be observed a slower convergence to the target points due to the saturation introduced to satisfy the actuators limitations. △

Experiment 2. The control law given in Example 2 has been used to let three robots patrol a given curve. The initial poses of the robots are $X_1(0) = -0.5\,\mathrm{m}$, $Y_1(0) = 0.5\,\mathrm{m}$, $\Theta_1(0) = \pi\,\mathrm{rad}$, $X_2(0) = 0\,\mathrm{m}$, $Y_2(0) = 0.5\,\mathrm{m}$, $\Theta_2(0) = \frac{\pi}{2}\,\mathrm{rad}$, $X_3(0) = 0.5\,\mathrm{m}$, $Y_3(0) = 0.5\,\mathrm{m}$, $\Theta_3(0) = \frac{3}{2}\pi\,\mathrm{rad}$.

Figure 7 reports some snapshots of the corresponding experiment; see [39] for the full video. As demonstrated by such a figure, the proposed control method



(a) Time $t = 0\,\mathrm{s}$.



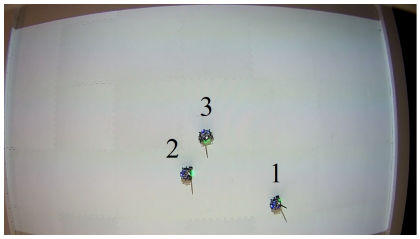
(b) Time $t = 10\,\mathrm{s}$.



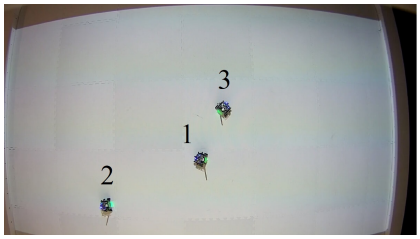
(c) Time $t = 20\,\mathrm{s}$.



(d) Time $t = 30\,\mathrm{s}$.



(e) Time $t = 40\,\mathrm{s}$.



(f) Time $t = 50\,\mathrm{s}$.

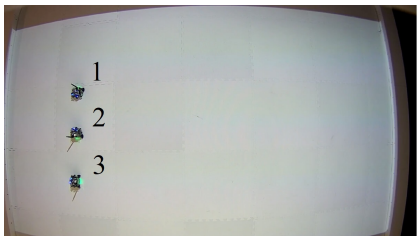
Figure 7: Result of the experiment with the control law given in Example 2.

let the three agents patrol the desired curve avoiding collisions among themselves and with fixed obstacles. However, note that the resulting trajectory is slightly different from the one depicted in Figure 4 due to position measurement errors, which, however, do not affect the convergence of the proposed navigation method by the robustness analysis reported at the end of Section 2. △

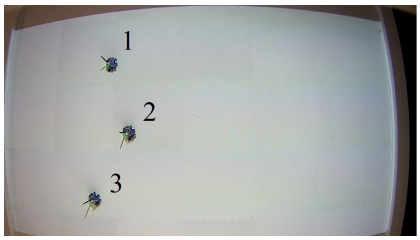
Experiment 3. The control law given in Example 3 has been used to let the leader patrol a circle while letting the other two robots be at a prescribed distance. The initial poses of the robots are $X_1(0) = -1\,\mathrm{m}$, $Y_1(0) = -0.6\,\mathrm{m}$, $\Theta_1(0) = 0\,\mathrm{rad}$, $X_2(0) = -0.5\,\mathrm{m}$, $Y_2(0) = 0\,\mathrm{m}$, $\Theta_2(0) = 0\,\mathrm{rad}$, $X_3(0) = -1\,\mathrm{m}$,

$Y_3(0) = 0.6\,\mathrm{m}$, $\Theta_3(0) = 0\,\mathrm{rad}$.

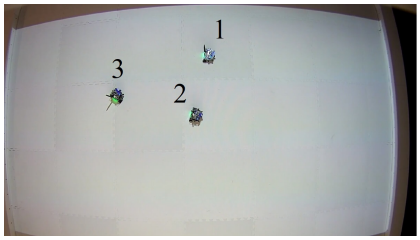
Figure 8 reports some snapshots of the corresponding experiment; see [40] for the full video. As shown by such a figure, the proposed control method



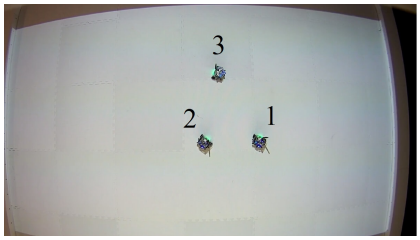
(a) Time $t = 0\,\mathrm{s}$.



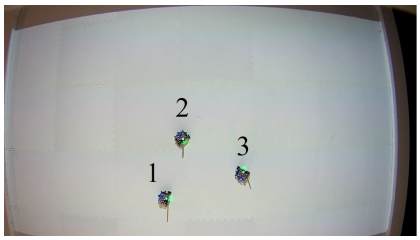
(b) Time $t = 10\,\mathrm{s}$.



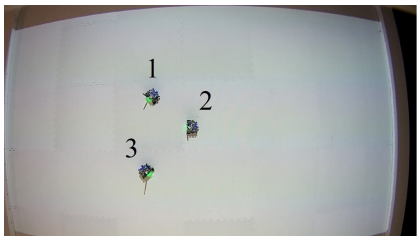
(c) Time $t = 20\,\mathrm{s}$.



(d) Time $t = 30\,\mathrm{s}$.



(e) Time $t = 40\,\mathrm{s}$.



(f) Time $t = 50\,\mathrm{s}$.

Figure 8: Result of the experiment with the control law given in Example 3.

allows the leader to patrol the desired curve while keeping the other agents at a prescribed distance and avoiding collisions among the robots and with fixed obstacles. △

6. Conclusions

A navigation strategy has been proposed to steer a team of unicycle-like mobile robots towards a desired path while maintaining a prescribed formation and avoiding collisions among the agents and with static obstacles in the environment. This goal has been reached by using some techniques borrowed from algebraic geometry to algorithmically construct a Lyapunov function that guarantees convergence to the desired formation and to the prescribed path if obstacles are absent. These techniques have been coupled with classical potential functions to guarantee that the path of motions of the agents are collision free, and with a hybrid mechanism that autonomously selects the navigation configurable parameters, thus making the overall procedure completely automatized.

In fact, given the prescribed path to be patrolled, the desired formation, the locations of the obstacles and the dimensions of the mobile robot, the results given in this work directly allows to algorithmically construct a navigation strategy that solves the problem, through the following steps:

1. Algorithm 1 is applied to determine functions f_b and g.
2. Equation (20) is used to determine η.
3. The formulas given in (24) are employed to compute V and ϱ.
4. Equations (7) and (8) are used to define the flow and the jump set.
5. The system (25) can be finally, directly implemented by selecting a suitable function Π (possibly with $\Pi(x, k, \kappa) = 1$, as suggested in Remark 2).

The robustness analysis inherited by the hybrid framework shows that the proposed navigation strategy is robust with respect to inaccurate measurements of the positions of the agents and imperfect implementation of the nominal strategy.

The effectiveness of the proposed motion planning strategy has been validated both in simulation and in experimental tests, carried out using a remotely accessible robotic testbed.

Future work will deal with the extension of the methods proposed in this paper to the decentralized case and to the presence of unknown obstacles in the workspace.

References

[1] J. Yu, S. M. LaValle, Optimal multirobot path planning on graphs: Complete algorithms and effective heuristics, IEEE Transactions on Robotics 32 (5) (2016) 1163–1177.

[2] B. Tang, K. Xiang, M. Pang, Z. Zhanxia, Multi-robot path planning using an improved self-adaptive particle swarm optimization, International Journal of Advanced Robotic Systems 17 (5) (2020).

[3] J.Yu, Average case constant factor time and distance optimal multi-robot path planning in well-connected environments, Autonomous Robots 44 (2020) 469–483.

[4] E. Rimon, D. E. Koditschek, Exact robot navigation using artificial potential functions, IEEE Transactions on Robotics and Automation 8 (5) (1992) 501–518.

[5] M. C. De Gennaro, A. Jadbabaie, Formation control for a cooperative multi-agent system using decentralized navigation functions, in: 2006 American Control Conference, Minneapolis, MN, USA, 2006.

[6] H. G. Tanner, A. Kumar, Towards decentralization of multi-robot navigation functions, in: 2005 IEEE International Conference on Robotics and Automation, Barcelona, Spain, 2005, pp. 4132–4137.

[7] H. G. Tanner, A. Boddu, Multiagent navigation functions revisited, IEEE Transactions on Robotics 28 (6) (2012) 1346–1359.

[8] H. Su, X. Wang, Z. Lin, Flocking of multi-agents with a virtual leader, IEEE Transactions on Automatic Control 54 (2) (2009) 293–307.

[9] C. Belta, V. Kumar, Abstraction and control for groups of robots, IEEE Transactions on Robotics 20 (5) (2004) 865–875.

[10] M. I. El-Hawwary, M. Maggiore, Distributed circular formation stabilization for dynamic unicycles, IEEE Transactions on Automatic Control 58 (1) (2013) 149–162.

[11] P. Tabuada, G. J. Pappas, P. Lima, Motion feasibility of multi-agent formations, IEEE Transactions on Robotics 21 (3) (2005) 387–392.

[12] N. E. Leonard, E. Fiorelli, Virtual leaders, artificial potentials and coordinated control of groups, in: 40th IEEE Conference on Decision and Control, Vol. 3, Orlando, FL, USA, 2001, pp. 2968–2973.

[13] H. G. Tanner, G. J. Pappas, V. Kumar, Leader-to-formation stability, IEEE Transactions on Robotics and Automation 20 (3) (2004) 443–455.

[14] A. Loria, J. Dasdemir, N. Alvarez Jarquin, Leader–follower formation and tracking control of mobile robots along straight paths, IEEE Transactions on Control Systems Technology 24 (2) (2016) 727–732.

[15] S. He, M. Wang, S. Dai, F. Luo, Leader–follower formation control of usvs with prescribed performance and collision avoidance, IEEE Transactions on Industrial Informatics 15 (1) (2019) 572–581.

[16] S. Dai, S. He, X. Chen, X. Jin, Adaptive leader–follower formation control of nonholonomic mobile robots with prescribed transient and steady-state performance, IEEE Transactions on Industrial Informatics 16 (6) (2020) 3662–3671.

[17] Z. Peng, G. Wen, S. Yang, A. Rahmani, Distributed consensus-based formation control for nonholonomic wheeled mobile robots using adaptive neural network, Nonlinear Dynamics 86 (2016) 605–622.

[18] C. Zong, Z. Ji, L. Tian, Y. Zhang, Distributed multi-robot formation control based on bipartite consensus with time-varying delays, IEEE Access 7 (2019) 144790–144798.

[19] K. K. Oh, M. C. Park, H. S. Ahn, A survey of multi-agent formation control, Automatica 53 (2015) 424–440.

[20] C. Possieri, M. Sassano, Patrolling and collision avoidance beyond classical navigation functions, in: European Control Conference, Limassol, Cyprus, 2018, pp. 1821–1826.

[21] C. Possieri, M. Sassano, Motion planning, formation control and obstacle avoidance for multi-agent systems, in: IEEE Conference on Control Technology and Applications, IEEE, Copenhagen, Denmark, 2018, pp. 879–884.

[22] M. Indri, C. Possieri, F. Sibona, P. D. C. Cheng, V. D. Hoang, Supervised global path planning for mobile robots with obstacle avoidance, in: 24th IEEE International Conference on Emerging Technologies and Factory Automation, Zaragoza, Spain, 2019, pp. 601–608.

[23] M. Indri, F. Sibona, P. D. C. Cheng, C. Possieri, Online supervised global path planning for AMRs with human-obstacle avoidance, in: 25th IEEE International Conference on Emerging Technologies and Factory Automation, Vol. 1, Vienna, Austria, 2020, pp. 1473–1479.

[24] S. Wilson, P. Glotfelter, L. Wang, S. Mayya, G. Notomista, M. Mote, M. Egerstedt, The robotarium: Globally impactful opportunities, challenges, and lessons learned in remote-access, distributed control of multirobot systems, IEEE Control Systems Magazine 40 (1) (2020) 26–44.

[25] R. Goebel, R. G. Sanfelice, A. R. Teel, Hybrid dynamical systems: modeling stability, and robustness, Princeton University Press, Princeton, NJ, 2012.

[26] D. A. Cox, J. Little, D. O'Shea, Ideals, varieties, and algorithms, Springer Science & Business Media, New York, 2015.

[27] D. A. Cox, J. Little, D. O'Shea, Using algebraic geometry, Springer Science & Business Media, New York, 2006.

[28] B. Siciliano, O. Khatib, Handbook of Robotics, Springer Verlag, 2008.

[29] F. Martinelli, C. Possieri, A. Tornambe, Motion planning for a unicycle-like mobile robot, using algebraic attractive curves, in: 22nd Mediterranean Conference on Control and Automation, IEEE, Palermo, Italy, 2014, pp. 1335–1340.

[30] T. Mylvaganam, M. Sassano, A. Astolfi, A differential game approach to multi-agent collision avoidance, IEEE Transactions on Automatic Control 62 (8) (2017) 4229–4235.

[31] H. K. Khalil, Nonlinear Systems, Prentice-Hall, New Jersey, 2002.

[32] T. Mylvaganam, C. Possieri, M. Sassano, Global stabilization of nonlinear systems via hybrid implementation of dynamic continuous-time local controllers, Automatica 106 (2019) 401–405.

[33] A. Seuret, C. Prieur, Event-triggered sampling algorithms based on a Lyapunov function, in: 50th IEEE Conference on Decision and Control, Orlando, FL, USA, 2011, pp. 6128–6133.

[34] R. Postoyan, P. Tabuada, D. Nesic, A. A. Martinez, A framework for the event-triggered stabilization of nonlinear systems., IEEE Transactions on Automatic Control 60 (4) (2015) 982–996.

[35] J. Chai, P. Casau, R. G. Sanfelice, Analysis and design of event-triggered control algorithms using hybrid systems tools, in: 56th IEEE Conference on Decision and Control, Melbourne, Australia, 2017, pp. 6057–6062.

[36] C. Possieri, A. Tornambe, On f-invariant and attractive affine varieties for continuous-time polynomial systems: the case of robot motion planning, in: 53rd IEEE Conference on Decision and Control, Los Angeles, CA, USA, 2014, pp. 3751–3756.

[37] C. Possieri, A. Tornambe, On polynomial vector fields having a given affine variety as attractive and invariant set: application to robotics, International Journal of Control 88 (5) (2015) 1001–1025.

[38] Experiment 1 Demo Video. Available online: `https://youtu.be/Ziz63_-BmqE` (accessed April 2021).

[39] Experiment 2 Demo Video. Available online: `https://youtu.be/Bc43Eh8PTf0` (accessed April 2021).

[40] Experiment 3 Demo Video. Available online: `https://youtu.be/Tt4CqQsyNi4` (accessed April 2021).

[41] R. T. Rockafellar, R. J.-B. Wets, Variational analysis, Springer, New York, 2009.

A. Proof of Theorem 1

The proof follows the same lines of the proof of [32, Thm. 1] with minor adaptations to consider the stability of set $\mathcal{V} \times \mathcal{A}$ rather than of the origin and to consider the fact that k is constant during flows. Letting $\xi = \mathrm{col}(x, k)$ and

$$F(\xi) = \begin{bmatrix} f(\xi) \\ 0 \end{bmatrix}, \qquad G(\xi) = \begin{bmatrix} x \\ \operatorname{argmin}_{\kappa \in \Xi(\xi)} \Pi(\xi, \kappa) \end{bmatrix},$$

the dynamics of the hybrid system (10) can be rewritten as

$$\dot{\xi} = F(\xi), \qquad \xi \in \mathcal{C}, \tag{28a}$$

$$\xi^+ \in G(\xi), \qquad \xi \in \mathcal{D}. \tag{28b}$$

We firstly show that system (28) is well-posed. Since, by assumption, the functions ℓ and ϱ are continuous, the sets $\mathcal{C}$ and $\mathcal{D}$ defined in (8) are closed. Furthermore, since f is continuous and hence locally bounded, the flow map $F(\xi)$ is outer semicontinuous and locally bounded for all $\xi \in \mathcal{C}$; see [41, Cor. 5.20]. On the other hand, under the hypotheses of the theorem, by [41, Ex. 5.22] the

set-valued mapping $\xi \rightrightarrows \operatorname{argmin}_{\kappa \in \Xi(\xi)} \Pi(\xi, \kappa)$ is outer semicontinuous and locally bounded, thus implying that also $G(\xi)$ is outer semicontinuous and locally bounded for all $\xi \in \mathcal{D}$. Finally, since $\Xi(\xi)$ is nonempty for all $\xi \in \mathcal{D}$ by Assumption 2, system (28) satisfies the "Hybrid Basic Conditions" stated in [25, Ass. 6.5] and hence it is well-posed by [25, Thm. 6.8].

Note that, by construction, if $k(0,0) \in \mathcal{A}$ then $k(t,j) \in \mathcal{A}$ for all $(t,j) \in \operatorname{dom}(k)$. Therefore, since $\mathcal{C} \cup \mathcal{D} = \mathbb{R}^n \times \mathcal{A}$ maximal solution of (28) starting in $\mathbb{R}^n \times \mathcal{A}$ are either complete or blow up in finite time. We can now prove the asymptotic stability of the set $\bar{\mathcal{V}} := \mathcal{V} \times \mathcal{A}$ for system (28). To this end, consider the Lyapunov function $\bar{V}(\xi) = V(x)$ that satisfies

$$\bar{V}(\xi) \geqslant \alpha_1(\|\xi\|_{\bar{\mathcal{V}}}), \qquad \forall \xi \in \mathcal{C} \cup \mathcal{D} \cup G(\mathcal{D}), \tag{29a}$$

$$\bar{V}(\xi) \leqslant \alpha_2(\|\xi\|_{\bar{\mathcal{V}}}), \qquad \forall \xi \in \mathcal{C} \cup \mathcal{D} \cup G(\mathcal{D}), \tag{29b}$$

$$\langle \nabla \bar{V}(\xi), F(\xi) \rangle \leqslant -\mu \rho(\xi), \qquad \forall \xi \in \mathcal{C}, \tag{29c}$$

$$\bar{V}(g) - \bar{V}(\xi) = 0, \qquad \forall \xi \in \mathcal{D}, g \in G(\xi), \tag{29d}$$

where $\rho(\xi) = \varrho(x)$ for all $\xi \in \mathcal{C}$, $\rho(\xi) \in \mathcal{PD}(\bar{\mathcal{V}})$. By (29c) and (29d), such a function is monotonically non-increasing along solutions to system (28), thus implying that its sub-level sets, which are compact by (29a), are positively invariant with respect to system (28). Thus, solutions to (28) starting in $\mathbb{R}^n \times \mathcal{A}$ are complete and the set $\bar{\mathcal{V}}$ is stable for system (28).

It remains to prove uniform convergence of solutions to (28) starting in $\mathbb{R}^n \times \mathcal{A}$ to the set $\bar{\mathcal{V}}$, that is for any $r > 0$ and $\varepsilon > 0$ there is $T > 0$ such that each solution to system (28) starting in $r\mathcal{B} \times \mathcal{A}$ satisfies $\|\xi(t,j)\|_{\bar{\mathcal{V}}} \leqslant \varepsilon$ for all $(t,j) \in \operatorname{dom}(\xi)$ such that $t + j \geqslant T$. Given $r > \varepsilon > 0$, let $c_0 > 0$ be such that $(r\mathcal{B} \times \mathcal{A}) \subset \mathcal{S}_0 := \{\xi \in \mathbb{R}^n \times \mathcal{A} : V(\xi) \leqslant c_0\}$ and let $c_1 > 0$ be such that $\mathcal{S}_1 := \{\xi \in \mathbb{R}^n \times \mathcal{A} : V(\xi) \leqslant c_1\} \subset (\varepsilon\mathcal{B} \times \mathcal{A})$. Note that these two constants exist by (29a) and (29b), respectively. Thus, letting ρ be such that (29c) holds, define

$$\vartheta := \inf_{\xi \in \mathcal{S}_0 \setminus \mathcal{S}_1} \rho(\xi),$$

which is a strictly positive constant since $\overline{\mathcal{S}_0 \setminus \mathcal{S}_1}$ is a compact set and $\bar{\mathcal{V}} \cap$

$(\overline{\mathcal{S}_0 \setminus \mathcal{S}_1}) = \emptyset$ due to the fact that $\bar{\mathcal{V}} \subset \mathcal{S}_1$. Furthermore, by considering that f is C^z for some sufficiently large $z \in \mathbb{N}$, there is a constant $\nu > 0$ such that

$$\langle \nabla(\ell(\xi) + \rho(\xi)), F(\xi) \rangle \leqslant \nu, \qquad \forall \xi \in \mathcal{S}_0 \setminus \mathcal{S}_1. \tag{30}$$

Therefore, define the constant

$$T := 1 + \frac{c_0(\vartheta(1-\mu)+\nu)}{\vartheta^2(1-\mu)\mu},$$

and assume by contradiction that there is a solution $\xi(t,j)$ to system (28) with $\xi(0,0) \in r\mathcal{B} \times \mathcal{A}$ that stays in $\mathcal{S}_0 \setminus \mathcal{S}_1$ for all $(t,j) \in \mathrm{dom}(\xi)$ such that $t + j \leqslant T$. Since $\xi^+ \in G(\xi)$, for all $h \in \mathbb{N}$, $h \geqslant 1$, such that $t_h + h \leqslant T$ it results that

$$\ell(\xi(t_h, h)) + \rho(\xi(t_h, h)) \leqslant 0. \tag{31}$$

Since $\rho(\xi) \geqslant \vartheta$ for all $\xi \in \mathcal{S}_0 \setminus \mathcal{S}_1$ and a jump occurs at hybrid time (t_{h+1}, h) only if $\xi(t_{h+1}, h) \in \mathcal{D}$, for all $h \in \mathbb{N}$, $h \geqslant 1$, such that $t_h + h \leqslant T$ it results that

$$\ell(\xi(t_{h+1}, h)) + \rho(\xi(t_{h+1}, h)) \geqslant (1-\mu)\rho(\xi(t_{h+1}, h)) \geqslant (1-\mu)\vartheta. \tag{32}$$

Therefore, the conditions given in (30), (31), and (32) imply that there is a minimum dwell time $\tau = \frac{(1-\mu)\vartheta}{\nu}$ between two consecutive jumps of the solution $\xi(t,j)$. Such a solution is therefore non-Zeno and $(t,j) \in \mathrm{dom}(\xi)$ implies $j \leqslant \frac{t}{\tau} + 1$. Hence, following [25, Thm. 3.18], by (29c) and (29d), we have

$$\bar{V}(\xi(t,j)) \leqslant \bar{V}(\xi(0,0)) - \mu\vartheta t \leqslant c_0 - \mu\vartheta \frac{\tau}{\tau+1}(t+j-1),$$

thus leading to a contradiction since $\xi \in \mathcal{S}_0 \setminus \mathcal{S}_1$ if and only if $c_1 < V(\xi) \leqslant c_0$. Thus, the set $\bar{\mathcal{V}}$ is globally uniformly asymptotically stable for system (28).

B. Proof of Proposition 1

Let the polynomials $p_{i,j} \in \mathbb{R}[x_i]$, $j = 1, \ldots, s_i$, $i = 1, \ldots, N$, be coerced into $\mathbb{R}[x]$ and consider the module $\mathcal{M}_i = \langle p_i, \ldots, p_{1,s_i} \rangle$ in $\mathbb{R}[x]$, $i = 1, \ldots, N$. By [27], one has that $\mathcal{V}_i = \mathbf{V}(\mathcal{M}_i)$, $i = 1, \ldots, N$, and $\mathcal{F} = \mathbf{V}(\langle q_1, \ldots, q_\omega \rangle)$.

Therefore, by [26], one has that the affine variety $\mathcal{V}$ given in (14) is given by

$$\begin{aligned}\mathcal{V} &= \mathbf{V}(\langle q_1, \ldots, q_\omega \rangle + \sum_{i=1}^{N} \mathcal{M}_i) \\ &= \mathbf{V}(\langle p_{1,1}, \ldots, p_{1,s_1}, p_{2,1}, \ldots, p_{2,s_2}, \ldots, p_{N,s_N}, q_1, \ldots, q_\omega \rangle).\end{aligned}$$

Thus, the statement follows by Algorithm 1 of [37].

C. Proof of Lemma 1

By Definition 3, a collision occurs if and only if there exists $t \in \mathbb{R}_{\geqslant 0}$ such that $x(t)$ belongs to the following variety

$$\left(\bigcup_{i=1}^{N} \bigcup_{w=1}^{W} \mathbf{V}(\varsigma_w(x_i)) \right) \cup \left(\bigcup_{i=1}^{N} \bigcup_{j=i+1}^{N} \mathbf{V}(c_{i,j}(x)) \right).$$

By [26], the affine variety above is given by $\mathbf{V}(\mathcal{M})$, where

$$\mathcal{M} = \left(\prod_{i=1}^{N} \prod_{w=1}^{W} \langle \varsigma_w(x_i) \rangle \right) \left(\prod_{i=1}^{N} \prod_{j=i+1}^{N} \langle c_{i,j}(x) \rangle \right).$$

Since each ideal in such a product is principal, then, by a trivial extension of [26, Ch. 4, §3, Prop. 6], the ideal $\mathcal{M}$ is principal and $\mathcal{M} = \langle b \rangle$. Therefore, a collision occurs at time $t \in \mathbb{R}_{\geqslant 0}$ if and only if $b(x(t)) = 0$.

D. Proof of Proposition 2

By computing the time derivative of $r(x)$ along the trajectories of system (21), one obtains that

$$\frac{d}{dt} r(x(t)) = \langle \nabla r(x(t)), -\eta(x(t))\beta(t) \rangle = -\eta^\top(x(t))\eta(x(t))\beta(t) \leqslant 0.$$

Therefore the function $r(x(t))$ is monotonically non-increasing in t. Hence, since $r(x(t)) \leqslant r(x_0) < \infty$ and $b(x(t)) = 0$ if and only if $r(x(t)) = \infty$, then there does not exist $t \in \mathbb{R}_{\geqslant 0}$ such that $b(x(t)) = 0$.

E. Proof of Theorem 2

Since $\lim_{x\to\partial\mathcal{R}} |\eta(x)| = \infty$ while $f_b(x)$ and $g(x)\zeta(x)$ are bounded, then $\gamma_1 b^2(x) f_b(x) + \gamma_1 b^2(x) g(x)\zeta(x) - \eta(x)\mu(x)p(x) \simeq -\eta(x)\mu(x)p(x)$ for all the points in the neighborhood of $\mathbf{V}(b) = \partial\mathcal{R}$. Thus, since $\mu(x)p(x) \geqslant 0$ for all $x \in \mathbb{R}^{2N} \setminus \mathcal{R}$, by the same reasoning used to prove Proposition 2, the set $\mathbb{R}^{2N} \setminus \mathcal{R}$ is positively invariant with respect to system (22), *i.e.*, no collision occurs. Hence, define $V = p^\top p$, whose time derivative is given by

$$\begin{aligned}\dot{V} &= -p^\top \gamma_1 b^2 \Lambda p - p^\top \tfrac{\partial p}{\partial x}\eta\mu p \\ &= -p^\top(\gamma_1 b^2\Lambda + \tfrac{\partial p}{\partial x}\eta\mu)p.\end{aligned}$$

Thus, the proof follows by classical Lyapunov arguments and by the fact that the set $\mathcal{Y}$ is positively invariant.

F. Proof of Theorem 3

Since the vector field $f(x,k)$ given in (23) and the functions $V(x)$ and $\varrho(x)$ given in (24) satisfy Assumptions 1 and 2 in the set $\mathcal{Y}$, then the hypotheses of Corollary 1 are met, thus implying that the hybrid implementation (10) solves the patrolling in formation problem for all $x_0 \in \mathcal{Y}$.

It remains to prove that the path of motions of the hybrid implementation (10) are collision free. Following the same reasoning employed in Appendix E, note that $f(x,k) \simeq -\gamma_2\eta(x)p^\top(x)p(x)$ for all the points in a neighborhood of $\partial\mathcal{R}$. Therefore, in such a neighborhood, one has that

$$\begin{aligned}\dot{r} &= -\gamma_2\eta^\top\eta p^\top p \leqslant -\underline{\gamma}_2\eta^\top\eta p^\top p, \\ r^+ &= r.\end{aligned}$$

where $\underline{\gamma}_2 = \min_{\mathcal{A}}\gamma_2 > 0$. Therefore, since there is a minimum dwell time between two consecutive jumps of the solution to the hybrid implementation (10) by the proof of Theorem 1, using the same reasoning used to prove Proposition 2, one concludes that the set $\mathbb{R}^{2N} \setminus \mathcal{R}$ is positively invariant with respect to system (10), *i.e.*, no collision occurs.