Automatica 173 (2025) 112047

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

A matching problem between two decoupled multi-agent systems with reference tracking capabilities ☆

Giuseppe Fedele [a], Luigi D'Alfonso [a,*], Boli Chen [b]

[a] University of Calabria, Department of Informatics, Modeling, Electronics and Systems Eng., Rende, Italy
[b] Department of Electronic and Electrical Engineering, University College London, London, UK

ARTICLE INFO

Article history:
Received 25 August 2023
Received in revised form 27 May 2024
Accepted 21 October 2024
Available online 19 December 2024

Keywords:
Multi-agent systems
Matching problem
Emergent behavior
Trajectory tracking and path following
Cooperative control

ABSTRACT

Multi-agent systems have emerged as the central framework in which autonomous agents are seamlessly coordinated and co-operate to address complex challenges. The area of matching between multiple swarms has attracted considerable attention, especially in critical domains such as transport, logistics and robotics, where coordination algorithms play an essential role in resource allocation, system performance and overall functionality. This paper addresses the challenges that arise in matching two swarms of agents, addressing issues related to scalability, dynamic environments, and privacy. The proposed solution, based on a theoretical study of emergent behavior in multi-agent systems, revolves around the design of a control law that aims to ensure the convergence of the agents' positions to a predefined region where the steady-state values remain independent of the agents' initial conditions. This independence reflects a collective behavior resulting from local interactions between the agents and their engagement with the environment. This property shows that even completely independent swarms can reach a coupled consensus within a predefined region, even in the absence of direct communication. The effectiveness of the proposed approach is supported by numerical simulations and experimental results.

1. Introduction

Multi-agent systems (MASs) have emerged as powerful paradigms for solving complex problems through the coordination and collaboration of multiple autonomous agents. In various fields such as transportation, logistics, robotics and economics, efficient and effective matching of agents has become an important area of research (Ren & Cao, 2011, Gazi & Passino, 2011). Matching between two or more MASs refers to the process of assigning agents from one system to agents in another system based on certain criteria or goals.

In 1962, David Gale and Lloyd Shapley were the first to introduce and solve the Stable Marriage Problem (Gale & Shapley, 1962) by proposing a sequential algorithm to establish matches between bipartite sets, considering scenarios such as matching students with their respective colleges or in marriage situations. Following the publication of their paper, various derivative forms of this foundational problem have been proposed and explored. Although the matching problem is usually formulated as an optimization problem, this paper takes an alternative approach. Given two swarms, each evolves independently and decoupled from the other to satisfy the conditions for formation. It is then shown that the solutions reached by the two MASs are equal. This guarantees the fulfillment of the matching condition as an emergent property of the proposed evolution model.

Literature review. Several solutions to the matching problem have been suggested within the optimization community. Notably, the most prevalent approaches are outlined in Gusfield and Irving (1989), Papadimitriou and Steiglitz (1998), which frame the matching problem as a combinatorial optimization challenge. Nevertheless, these methods exhibit two shortcomings. Firstly, they are centralized approaches, meaning that comprehensive information, encompassing the states of all agents, must be gathered on a server, potentially leading to a substantial computational burden for large-scale systems. Secondly, these techniques are exclusively suitable for static systems. Consequently, if there are alterations in the number of agents or environments, the problem necessitates being readdressed.

Matching techniques find extensive applications in the field of robotics, facilitating coordination and cooperation among multiple robots (Khamis, Hussein, & Elmogy, 2015). In general, two crucial applications of matching algorithms are task assignment and object manipulation, as discussed in Caccavale and Uchiyama

☆ The material in this paper was not presented at any conference. This paper was recommended for publication in revised form by Associate Editor Keyou You under the direction of Editor Christos G. Cassandras.

* Corresponding author.
E-mail addresses: giuseppe.fedele@unical.it (G. Fedele), luigi.dalfonso@unical.it (L. D'Alfonso), boli.chen@ucl.ac.uk (B. Chen).

https://doi.org/10.1016/j.automatica.2024.112047

(2016), Smith and Bullo (2009). In task assignment, algorithms optimize by considering individual robot capabilities and task requirements, ensuring efficient utilization of the robot workforce based on skills, availability, and task complexity. In object manipulation, the collaborative approach of matching enables multiple robots to work synchronously, particularly when paired with complementary skills. This addresses tasks that might be too challenging for a single robot due to complexity or difficulty. A further area where the effective matching plays a significant role is formation control, as described in Sakurama, Azuma, and Sugie (2018). In this case, the desired arrangement and coordination of multiple robots is achieved by assigning them to specific roles or positions in a formation while simultaneously ensuring synchronized movement and maintaining the desired geometric configuration. This capability is particularly useful in applications where precise coordination between robots is required (Fedele, D'Alfonso, Bono, & Gazi, 2023). In the context of positioning and mapping of multiple robots, matching algorithms help with simultaneous localization and mapping tasks by matching sensor data from different robots to create a coherent map of the environment while estimating the relative positions of the robots (Pinto, Sobreira, Moreira, Mendonça, & Matos, 2013). Matching algorithms are also important in the context of robot swarms, where collective behavior emerges from the interactions of individual robots (Majid, Arshad, & Mokhtar, 2022), and they assign specific roles or functions to robots within the swarm, enabling coordinated behavior and the achievement of desired swarm goals.

When looking at specific cases of matching between robots or vehicles, three possible examples can be considered. In the context of cyclic transport with agent pairs, two autonomous vehicle swarms, labeled as SW_1 and SW_2, engage in a systematic transport of goods between different points of interest. Agents within these swarms form pairs cyclically. This matching process is recognized for its potential to optimize the overall efficiency of the transport system, a notion extensively discussed in Farivarnejad, Wilson, and Berman (2016) and Unhelkar et al. (2018). Moving to the second scenario, the domain of intelligent transportation within a network of autonomous vehicles on a shared highway can be examined. Here, agents from fleets SW_1 and SW_2 collaborate at intersections, utilizing coordination strategies to optimize traffic flow. Pairs of vehicles align in a straight line, facilitating seamless transitions and alleviating congestion issues. Detailed insights into this orchestrated movement can be found in studies such as Johansson, Mårtensson, Sun, and Yin (2021) and Firoozi, Zhang, and Borrelli (2021). As a third scenario, pipeline inspection, where ground-based robots in SW_1 and aerial drones in SW_2 collaborate to inspect extensive pipeline networks, can be considered. Coordinated movement ensures synchronized data collection from both ground and aerial perspectives, enhancing the inspection process. This collaborative approach is outlined in studies like Meng et al. (2020), Queralta et al. (2020), and Tian, Chen, Sagoe-Crentsil, Zhang, and Duan (2022). In each of these examples, the concept of matching assumes a crucial role in optimizing processes, improving efficiency, and facilitating seamless collaboration among autonomous entities.

More recently, in Watanabe and Sakurama (2022) a distributed dynamic matching is considered where agents autonomously determine their matching partners while moving with local information. The solution proposed by the authors tends to overcome the drawbacks of previous works, as it does not suffer from local minima since agents and targets can move in the matching problem; moreover, it does not require indexing of agent and communication between agents but only relative position detection. In the above scenarios, coordinating multi-agent systems poses challenges arising from inherent complexity and system diversity, as highlighted in Ren and Cao (2011). These challenges, categorized into scalability, dynamic environment and privacy, must be addressed to successfully implement matching algorithms in multi-agent systems.

Main contribution. This paper deals with the coordination problem between two swarms of agents, where each agent is modeled as a single integrator in a two-dimensional space. The goal is to develop control protocols for both swarms that allow the agents to traverse certain paths or shapes while reaching a pairwise consensus between the two swarms. In essence, both MASs adhere to predefined motion profiles on prescribed paths, ensuring that each agent in the first swarm achieves consensus with precisely one agent in the second swarm, thereby establishing meaningful agent matches. The main contributions of this work are to design control laws that satisfy the following criteria:

- the agents in each swarm fulfill the conditions for approaching the desired paths in finite-time, while the agents' positions converge within a predefined boundary, with respect to the chosen movement profile;
- a pairwise consensus is reached between the agents of the two swarms;
- the coordination problem is solved, taking into account the decoupling of the two swarms and the lack of communication between them.

The proposed solution is rooted in a theoretical examination of emergent behavior within the specified category of multi-agent systems. Specifically, the designed control law aims to ensure the convergence of agents' positions to a predetermined region. In this region, steady-state values are independent of initial agent conditions, reflecting a collective behavior emerging from local interactions among agents and their interactions with the environment. This property is significant as it demonstrates that entirely independent swarms can achieve a coupled consensus within a predefined region, even without direct communication between agents and regardless of their initial conditions. To the best of our knowledge, this behavior represents a novel development in the field of multi-agent systems formation, marking an innovation compared to the existing state of the art. Additionally, this decoupling decision takes privacy into account, especially in applications where agents have private information that should not be revealed during matching. By keeping the two swarms decoupled, the problem creates a certain level of privacy and ensures that confidential information remains protected. The proposed solution to the described problem also effectively addresses the challenges related to scalability and dynamic real-world scenarios in the area of matching between two swarms of agents. The chosen model for the agents is a kinematic model, disregarding the individual agent's dynamics. This seemingly simplistic choice has demonstrated efficacy in practical situations as a reference trajectory for real robots, especially in the case of certain flying robots and all flat systems, as discussed in Fliess, Lévine, Martin, and Rouchon (1995). This approach proves effective provided that at least one point of the robot body can asymptotically follow any smooth trajectory, as outlined in Franchi, Stegagno, and Oriolo (2016). For a comprehensive exploration of the extensive potential of kinematic models, refer to Zhao and Sun (2017) and the associated literature. The problem acknowledges the scalability challenge by designing the control protocols to allow the movement of agents in a bounded region around the reference motion profile, in finite-time, with computational complexity independent of the number of agents, since the solution is represented as an emergent behavior of the multi-agent model, as in the case of Fedele, D'Alfonso, and Bono (2023). The matching problem also takes into account the dynamic nature of real-world scenarios. Agents operate in environments where they must move along a

trajectory according to a motion profile. The control protocols are designed to adapt and respond to these dynamic conditions to ensure the validity of the matching assignments. By continuously adapting the agents' movements along the desired paths, the problem addresses the challenges posed by changing real-world scenarios.

The paper is structured as follows: Section 2 contains the formulation of the matching problem to be solved. Section 3 presents the properties of the diffeomorphism used to transform the problem into a framework in which a simple solution can be found. The theoretical properties of the proposed solution are described in Section 4. Section 5 considers the case of more than two multi-agent systems. Section 6 provides numerical simulations and experimental results to demonstrate the effectiveness of the proposed approach. The final section concludes the paper and discusses possible avenues for future research.

Notations and preliminaries

The following notation will be used throughout the paper. $\mathbb{R}_{\geq 0}$ (resp. $\mathbb{R}_{>0}$) is the set of nonnegative (resp. positive) real numbers. The set of positive integer numbers is denoted as $\mathbb{N}^+$. The sign function is $\text{sign}(m) = \{1, \text{ if } m \geq 0, \ 0, \text{ otherwise}\}$. A swarm of n agents is modeled as a graph $\mathcal{G} = (V, E)$ where $V = \{1, \dots, n\}$ is the set of nodes and $E \subseteq \{(i, j) : i, j \in V, i \neq j\}$ (no self-loops are allowed) is the set of unordered pairs of vertices defining the edges of the graph. Agents are the vertices of the graph while edges indicate the pair of agents that have interactions. The set of neighbors of agent i is indicated by $\mathcal{N}_i = \{j \mid (i, j) \in E\}$ and represents the set of agents that individual i interacts with. The graph $\mathcal{G}$ is complete if there is an edge for each pair of nodes, i.e. $\mathcal{N}_i = \{1, \dots, n \setminus i\}$, $\forall i = 1, \dots, n$. Given a parametric curve $\gamma : \mathbb{R} \to \mathbb{R}^2$ defined in Cartesian coordinates as $\gamma(r) = \left[\gamma_x(r), \gamma_y(r)\right]^T$ with the variable $r \in \mathbb{R}$ defining its parametrization, let $\overline{\gamma}(r)$ and $\underline{\gamma}(r)$ be two curves representing an upward and downward shifted version of the $\gamma(r)$, respectively. In this work, the shifted versions of $\gamma(r)$ are defined as $\overline{\gamma}(r) = [\gamma_x(r), \gamma_y(r) + \overline{\alpha}]^T$, $\underline{\gamma}(r) = [\gamma_x(r), \gamma_y(r) - \underline{\alpha}]^T$ with $\overline{\alpha}, \underline{\alpha} \in \mathbb{R}_{\geq 0}$ in the case of open curves, while $\overline{\gamma}(r) = \overline{\alpha}\left(\gamma(r) - c_\gamma\right) + c_\gamma$, $\underline{\gamma}(r) = \underline{\alpha}\left(\gamma(r) - c_\gamma\right) + c_\gamma$ where c_γ is the centroid of γ and $\overline{\alpha} \geq 1, \underline{\alpha} \in (0, 1]$, in the case of closed curves.

2. Problem formulation

Consider two swarms of $n \in \mathbb{N}^+$ agents, namely $\bar{s}$ and $\underline{s}$, modeled as single integrators in $\mathbb{R}^2$:

$$\begin{aligned} \dot{\bar{s}}_i(t) &= \bar{u}_{s,i}(t), \\ \dot{\underline{s}}_i(t) &= \underline{u}_{s,i}(t), \quad i = 1, \dots, n, \end{aligned} \tag{1}$$

where $\bar{s}_i(t)$, $\underline{s}_i(t)$ represent the position of the ith agent of the two swarms, $\bar{u}_{s,i}(t), \underline{u}_{s,i}(t) \in \mathbb{R}^2$ are the control protocols to be designed.

Assumption 1. the graphs encoding the interactions in $\bar{s}$ and $\underline{s}$ are complete, and there is an absence of interconnection between the two swarms. In other words, the two sets of agents are independent of each other, with no communication occurring between agents from different swarms; agents from one swarm are unable to measure their displacement to agents belonging to another swarm.

While the assumption of a complete connection may seem to pose a considerable drawback to the feasibility of the proposed solution, it is conceivable that exploiting partial connectivity could be a viable alternative for guiding agents within a confined space. Within this confined area, the onboard sensors of each agent could facilitate communication with all others, thus meeting the requirement of complete connectivity. This potential drawback could, therefore, be mitigated by harnessing partial connectivity to direct agents effectively within the designated region (Bono, D'Alfonso, Fedele, & Gazi, 2022).

Consider a continuous and differentiable curve $\gamma \in \mathcal{C}^2$, with $\gamma : \mathbb{R} \to \mathbb{R}^2$, representing a given path or shape defined in Cartesian coordinates as $\gamma(\omega) = \left[\gamma_x(\omega), \gamma_y(\omega)\right]^T$ where the variable $\omega \in \mathbb{R}$ is related to its parametrization. More precisely, a motion profile on γ can be chosen by describing the evolution over time, namely $\omega(t)$, and looking at the resulting value of the curve, i.e. $\gamma(\omega(t))$. It is worth noting that the curve $\gamma(\omega)$ may represent a target path along which the agents of the two swarms must couple to perform certain operations by moving in a coordinated manner and following the profile defined by the time law $\omega(t)$.

Let $\overline{\gamma}(\omega)$ and $\underline{\gamma}(\omega)$ be two curves representing an upward and downward shifted version of the $\gamma(\omega)$, respectively.

Assumption 2. the agents of the swarm $\bar{s}_i$ (or $\underline{s}_i$) have information about the curve $\overline{\gamma}(\omega)$ (or $\underline{\gamma}(\omega)$) and are able to measure their relative displacements from it.

The above information on $\overline{\gamma}(\omega)$ and $\underline{\gamma}(\omega)$ is used explicitly for control purposes by adjusting the formation of each swarm accordingly so that the desired fit (tangential or radial, as described later) is achieved. The entire control protocol corresponds to the *map-design-map back scheme*. First, the MAS is mapped into a virtual reference frame that utilizes the information of the path $\gamma(\omega)$ on which the agents must be driven, taking into account position errors with respect to the reference path and movement errors with respect to a given target movement profile $\omega(t)$. In this novel reference frame, a virtual control law is designed in a simpler way, which is then fed back into the real reference frame to control the real multi-agent system. Going deeper, a time-varying diffeomorphism $\mathcal{D}_{r2v}^{(\gamma)}(\cdot, \cdot)$ is designed to directly connect the real reference frame where the agents move to a virtual reference frame related to the curve γ. In other words, $\mathcal{D}_{r2v}^{(\gamma)} : \mathbb{R}_{\geq 0} \times \mathbb{R}^2 \to \mathbb{R}^2$ uniquely maps a point $\tilde{p} \in \mathbb{R}^2$ at a given time instant $\tilde{t} \geq 0$ and considering a shape $\tilde{\gamma}$, compliant with the given assumptions, into a point $\tilde{v} = \mathcal{D}_{r2v}^{(\tilde{\gamma})}(\tilde{t}, \tilde{p})$. The definition of the diffeomorphism is strictly related to the context of interest and depending on it the here following matching problem will assume a different meaning and will ensure different properties, as it will be clarified in the two proposed instances for the mapping.

If the diffeomorphism is structured as $\mathcal{D}_{r2v}^{(\gamma)}(\cdot, \cdot) = \left[\mathcal{D}_{r2v}^{(\gamma,1)}(\cdot, \cdot), \mathcal{D}_{r2v}^{(\gamma,2)}(\cdot, \cdot)\right]^T$, i.e. $\mathcal{D}_{r2v}^{(\gamma,k)}(\cdot, \cdot)$, $k = 1, 2$ represents the function that maps into the first axis and second axis of the virtual reference frame, respectively, the problem to address can be stated as follows.

MATCHING PROBLEM (MP)

Design the control protocols $\bar{u}_{s,i}(t)$ and $\underline{u}_{s,i}(t)$ such that the following requirements are fulfilled:

- R_1: the second virtual component $\mathcal{D}_{r2v}^{(\overline{\gamma},2)}(t, \bar{s}_i(t))$ of each agent $\bar{s}_i(t)$ (resp. $\mathcal{D}_{r2v}^{(\underline{\gamma},2)}(t, \underline{s}_i(t))$ and $\underline{s}_i(t)$) move in finite-time to 0;
- R_2: for each $\bar{s}_i$ there exists one and only one $\underline{s}_j$, such that $\lim_{t\to\infty}\left[\mathcal{D}_{r2v}^{(\overline{\gamma},1)}(t, \bar{s}_i(t)) - \mathcal{D}_{r2v}^{(\underline{\gamma},1)}(t, \underline{s}_j(t))\right] = 0$;
- R_3: agents organise so that there exists $\Gamma > 0$ such that $\lim_{t\to\infty}\|\bar{s}_i(t) - \overline{\gamma}(\omega(t))\| \leq \Gamma$, $\lim_{t\to\infty}\|\underline{s}_i(t) - \underline{\gamma}(\omega(t))\| \leq \Gamma$, $i = 1, \dots, n$.

In other words, the main idea is to make zero the component of each agent on one of the virtual axes while simultaneously ensuring that on the other axis, a matching exists between each agent of the first swarm with a single agent of the second one. This type of problem can alternatively be formulated in a constrained optimization framework, in which the individuals of the MASs tend to match each other pairwise by minimizing the following cost at each time instant

$$\sum_{i=1}^{n} \left(\mathcal{D}_{r2v}^{(\overline{\gamma},2)}(t,\bar{s}_i(t))\right)^2 + \left(\mathcal{D}_{r2v}^{(\underline{\gamma},2)}(t,\underline{s}_i(t))\right)^2,$$

with constraints imposed by R_2 and R_3. Instead of facing this constrained optimization directly, this article proposes a decoupled solution in which each MAS solves its own problem with respect to R_1 and R_3, while the two decoupled swarms satisfy the requirement R_2 as an emergent property.

To give a practical idea on how to design the diffeomorphism and how the defined matching problem can be formulated in real contexts, two main scenarios will be hereafter described.

Remark 1. Note that the case of $\overline{\alpha} = \underline{\alpha} = 1$ for closed curves, or $\overline{\alpha} = \underline{\alpha} = 0$ for open curves, implies that the two reference curves are the same. In this case, the matching problem translates into a *paired consensus*, i.e., a consensus for couple of agents. In other words, for each agent in the first swarm, there will be one and only one agent in the second multi-agent system that will reach a consensus on the same state of the first one.

2.1. Matching on the tangent of a curve

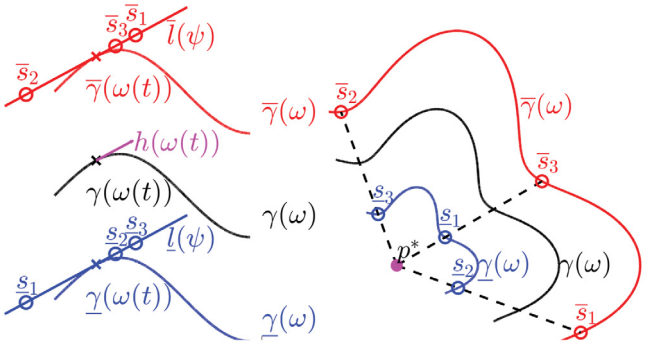
In the realm of modern logistics and transportation, efficient cargo handling and delivery are paramount. To optimize the movement of goods, it becomes necessary for two distinct groups of vehicles to position themselves at a fixed distance from one another. The goal is to facilitate the formation of collaborative caravans, wherein pairs of vehicles from different teams work and move in tandem to transport cargo effectively. In particular, the synchronized positioning of a couple of vehicles ensures that loads are evenly distributed in the pair and this optimal utilization of each vehicle's capacity maximizes the total cargo that can be transported, streamlining logistics operations. In this context, given a reference curve γ, that can represent for instance a road where vehicles should move, and the two defined upward and downward shifted versions that stand for the virtual roads for each of the two multi-agent systems, the control law for each agent can be designed so that to drive it on the tangent of the reference profile simultaneously ensuring the MP is solved. In other words, consider the unit vector

$$h(\omega(t)) = \left.\frac{\frac{d\gamma(\omega)}{d\omega}}{\left\|\frac{d\gamma(\omega)}{d\omega}\right\|}\right|_{\omega(t)}$$

tangent to γ in $\gamma(\omega(t))$ and the projection operator on it, i.e. $\Pi(\omega(t)) = h(\omega(t))h(\omega(t))^T$. The MP translates as follows.

Matching Problem in the tangent case.
Design the control protocols $\overline{u}_{s,i}(t)$ and $\underline{u}_{s,i}(t)$ such that:

- agents $\bar{s}_i(t)$ (resp. $\underline{s}_i(t)$), $i = 1, \dots, n$, move in finite-time on the line $\bar{l}(\psi) = \overline{\gamma}(\omega(t)) + h(\omega(t))(\psi - \omega(t))$ (resp. $\underline{l}(\psi) = \underline{\gamma}(\omega(t)) + h(\omega(t))(\psi - \omega(t))$);
- for each $\bar{s}_i$ there exists one and only one $\underline{s}_j$, such that
  $\lim_{t\to\infty} \left[\Pi(\omega(t))\,(\bar{e}_i(t)) - \Pi(\omega(t))\,(\underline{e}_j(t))\right] = 0,$
  where $\bar{e}_i(t) = \bar{s}_i(t) - \overline{\gamma}(\omega(t))$, $\underline{e}_j(t) = \underline{s}_j(t) - \underline{\gamma}(\omega(t))$;



(a) Matching Problem in the tangent case: the agents of the two multi-agent systems go on the tangent lines in the points $\overline{\gamma}(\omega(t))$ and $\underline{\gamma}(\omega(t))$.

(b) Matching Problem in the radial case: the agents of the two multi-agent systems go on the curves $\overline{\gamma}$ and $\underline{\gamma}$.

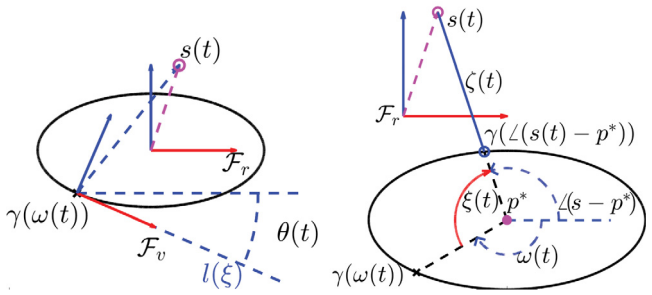
Fig. 1. Matching Problems in the tangent and radial cases.

- agents organize such that
  $$\lim_{t\to\infty} \|\bar{e}_i(t)\| \le \Gamma, \quad \lim_{t\to\infty} \|\underline{e}_i(t)\| \le \Gamma, \quad i = 1, \dots, n.$$

Fig. 1(a) illustrates the problem to be solved. In particular, once we have chosen the desired trajectory $\gamma(\omega)$ (black curves), the profile $\omega(t)$ and the corresponding curves $\overline{\gamma}(\omega)$ and $\underline{\gamma}(\omega)$ (red and blue curve) at each point of the motion profile, more precisely $\overline{\gamma}(\omega(t))$ and $\underline{\gamma}(\omega(t))$, it is possible to define the tangents to the two curves $\bar{l}(\psi)$ and $\underline{l}(\psi)$ on which the agents of the two swarms must position themselves. The proposed matching problem requires that the agents of the two swarms arrange themselves in such a way that for each agent of the first swarm there is one and only one agent of the second swarm, so that the relative positions with respect to the motion profile on the corresponding tangent are equal. In the considered figure, the agents $\bar{s}_1$ and $\underline{s}_3$ are matched, as is $\bar{s}_2$ with $\underline{s}_1$ and $\bar{s}_3$ and $\underline{s}_2$. The agents then maintain these relative positions during their movement according to the motion profile.

2.2. Matching in a radial way

In this second case, the main idea is to start from a closed curve and to ensure that agents are able to synchronize their rotational speeds. In detail, given a closed curve γ that assumes a radial parametrization w.r.t. a given internal vantage point $p^* \in \mathbb{R}^2$ (as in the case of a star-shaped set, see Rimon & Koditschek, 1991) and its upward and downward shifted versions, the two MASs have to place themselves on $\overline{\gamma}$ and $\underline{\gamma}$, respectively, while simultaneously ensuring that for each agent on the internal curve $\underline{\gamma}$ there exists one and only one agent on the external path $\overline{\gamma}$ that moves with the same angle w.r.t. p^*. By enforcing synchronized rotational speeds, two coupled vehicles move according to the same angular velocity along their respective curves. This conservation of angular velocity plays a crucial role in preserving the congruence of the curves' shapes and, consequently, the proportional relationship between the vehicles' positions. As the curves are of equal shape, maintaining the same rotational speed guarantees that both vehicles experience identical curvilinear motion. This synchrony allows the vehicles to traverse corresponding distances along their respective curves simultaneously, maintaining a consistent and predictable alignment. The concept of synchronized motion along scaled curves finds significant applications in robotics and automation. In multi-robot systems or autonomous



(a) Proposed diffeomorphism in the case of a tangent mapping and of an ellipse as reference curve γ. The point $s(t)$ (magenta void circle) is defined by the magenta dashed vector in the real world and it is mapped into the blue dashed vector in the virtual frame.

(b) Proposed diffeomorphism in the case of a polar mapping and of an ellipse as reference curve γ. The point $s(t)$ is defined by the magenta dashed vector in the real world and it is mapped to the virtual reference frame by the angle $\xi(t)$ and the distance $\zeta(t)$. These values represent the coordinates of $x(t)$ in $\mathcal{F}_v$.

Fig. 2. Proposed diffeomorphisms.

vehicles, this approach enables seamless coordination, facilitating complex tasks and missions with enhanced precision and safety. The MP translates as follows.

Matching Problem in the polar case.
Design the control protocols $\overline{u}_{s,i}(t)$ and $\underline{u}_{s,i}(t)$ such that:

- agents $\overline{s}_i(t)$ (resp. $\underline{s}_i(t)$), $i = 1, \dots, n$, move in finite-time on $\overline{\gamma}(\omega(t))$ (resp. $\underline{\gamma}(\omega(t))$);
- for each $\overline{s}_i$ there exists one and only one $\underline{s}_j$, such that $\lim_{t\to\infty}\left[\angle\left(\overline{s}_i - p^*\right) - \angle\left(\underline{s}_j - p^*\right)\right] = 0$;
- agents organize so that $\lim_{t\to\infty}\|\overline{s}_i(t) - \overline{\gamma}(\omega(t))\| \le \Gamma$, $\lim_{t\to\infty}\|\underline{s}_i(t) - \underline{\gamma}(\omega(t))\| \le \Gamma$, $i = 1, \dots, n$.

Fig. 1(b) illustrates the problem to be solved in this second case. Given the desired trajectory $\gamma(\omega)$ (black curve), the profile $\omega(t)$ and the corresponding curves $\overline{\gamma}(\omega)$ and $\underline{\gamma}(\omega)$ (red and blue curves), the available parametrization ensures that a radial representation for each agent can be given. The proposed matching problem requires that the agents of the two swarms arrange themselves in such a way that for each agent of the first swarm there is one and only one agent of the second swarm that is radially aligned with the first one. In the considered figure, the agents $\overline{s}_1$ and $\underline{s}_2$ are matched, as is $\overline{s}_2$ with $\underline{s}_3$ and $\overline{s}_3$ and $\underline{s}_1$. The agents then maintain these relative positions during their movement according to the motion profile.

3. Reference frames mapping

The key to solving MP lies in the use of a mapping between the reference frame in which the multi-agent system is defined and a new frame in which the errors between the states of the agents and the target curve are taken into account together with a tracking error with respect to the motion profile $\omega(t)$. More precisely, let $\mathcal{F}_r$ be the actual planar reference frame in which the system (1) evolves, and let $\mathcal{F}_v$ be the novel virtual frame accounting the position of the agent on the curve. From a general point of view, let us consider a class of nonlinear systems of the form

$$\dot{s} = g(t, s) \tag{2}$$

defined in $\mathcal{F}_r$ with $t \in \mathbb{R}_{\ge 0}$, $s \in \mathbb{R}^2$ and $g(\cdot,\cdot) : \mathbb{R}_{\ge 0} \times \mathbb{R}^2 \to \mathbb{R}^2$ for which there is a time-varying change in the variables

$$x = \mathcal{D}^{(\gamma)}_{r2v}(t, s) \tag{3}$$

such that

$$\frac{\partial \mathcal{D}^{(\gamma)}_{r2v}(t,s)}{\partial s} g(t,s) = f(x) - \frac{\partial \mathcal{D}^{(\gamma)}_{r2v}(t,s)}{\partial t}, \tag{4}$$

where $f(\cdot) : \mathbb{R}^2 \to \mathbb{R}^2$.

Proposition 2. *If $\mathcal{D}^{(\gamma)}_{r2v}(t, s)$ is a diffeomorphism then the change of variable converts the system (2) into the equivalent time-invariant system $\dot{x} = f(x)$.*

Proof. From Eq. (3) it follows that

$$\dot{x} = \frac{\partial \mathcal{D}^{(\gamma)}_{r2v}(t,s)}{\partial t} + \frac{\partial \mathcal{D}^{(\gamma)}_{r2v}(t,s)}{\partial s}\dot{s}.$$

Since $\mathcal{D}^{(\gamma)}_{r2v}(t, s)$ is a diffeomorphism, $\frac{\mathcal{D}^{(\gamma)}_{r2v}(t,s)}{\partial s}$ is nonsingular for all s and $g(t, s)$ is (Plastock, 1974):

$$\dot{s} = g(t,s) = \left[\frac{\partial \mathcal{D}^{(\gamma)}_{r2v}(t,s)}{\partial s}\right]^{-1}\left(f(x) - \frac{\partial \mathcal{D}^{(\gamma)}_{r2v}(t,s)}{\partial t}\right), \tag{5}$$

where $\frac{\partial \mathcal{D}^{(\gamma)}_{r2v}(t,s)}{\partial s}$ is the Jacobian of the diffeomorphism computed in the current (t, s). □

Proposition 2 is therefore useful to convert the solution obtained in the virtual reference frame to the actual one. Indeed, given the two multi-agent systems $\overline{s}$ and $\underline{s}$ modeled as (1), the overall solution consists of these three main steps:

- transform $\overline{s}_i(t), \underline{s}_i(t)$, $i = 1, \dots, n$ respectively into related virtual agents $\overline{x}_i(t), \underline{x}_i(t)$, $i = 1, \dots, n$ by using the diffeomorphisms $\mathcal{D}^{(\overline{\gamma})}_{r2v}(\cdot,\cdot)$ and $\mathcal{D}^{(\underline{\gamma})}_{r2v}(\cdot,\cdot)$. The following models are obtained in the virtual frame
$$\dot{\overline{x}}_i(t) = \overline{u}_{x,i}(t),$$
$$\dot{\underline{x}}_i(t) = \underline{u}_{x,i}(t), \ i = 1, \dots, n; \tag{6}$$
- design the control laws $\overline{u}_{x,i}(t)$ and $\underline{u}_{x,i}(t)$ to solve MP in the virtual frame;
- transform the virtual solutions back into the real ones using Proposition 2, i.e. applying (5) to $\overline{u}_{x,i}(t)$, $\underline{u}_{x,i}(t)$ and obtaining $\overline{u}_{s,i}(t)$, $\underline{u}_{s,i}(t)$, $\forall i$, respectively.

Considering the two defined matching examples, two different diffeomorphic mapping $\mathcal{D}^{(\gamma)}_{r2v}(\cdot,\cdot)$ have to be designed, one for each of the cases of interest.

3.1. Mapping in the tangent case

In the case of matching on the tangent to the curve, as described in Section 2.1, the diffeomorphism $\mathcal{D}^{(\gamma)}_{r2v}(t, s)$ proposed in this paper can be defined as follows. Let us consider the error signal (see Fig. 2(a)) $e(t) \triangleq s(t) - \gamma(\omega(t))$ and let $\theta(t)$ be the angle between the tangent at $\gamma(\omega(t))$ and the first axis of $\mathcal{F}_r$, that is $\theta(t) = \angle h(\omega(t))$.

Then the time-varying coordinates transformation $\mathcal{D}_{r2v}^{(\gamma)}(t,s)$ is computed as

$$\mathcal{D}_{r2v}^{(\gamma)}(t,s) = R(t)e(t), \tag{7}$$

where

$$R(t) = \begin{bmatrix} \cos(\theta(t)) & \sin(\theta(t)) \\ -\sin(\theta(t)) & \cos(\theta(t)) \end{bmatrix}. \tag{8}$$

Given a position $s(t)$ in the real frame $\mathcal{F}_r$ then, according to (7), (8) and Proposition 2, it follows that

$$\frac{\partial \mathcal{D}_{r2v}^{(\gamma)}(t,s)}{\partial s} = R(t),$$

$$\frac{\partial \mathcal{D}_{r2v}^{(\gamma)}(t,s)}{\partial t} = \dot{R}(t)e(t) - R(t)\frac{\partial \gamma(\omega(t))}{\partial \omega(t)}\dot{\omega}(t),$$

where, as shown in Fedele, D'Alfonso, and Gazi (2022), $\dot{R}(t) = \dot{\theta}(t)R(t)A$ with $A = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$. In the new frame $\mathcal{F}_v$, it is possible to design a control law by decoupling its action along the two axes. More precisely, since it is necessary that agents distribute themselves along the first axis, the control action must cancel the component along the second axis and guarantee the desired behavior along the first one.

3.2. Mapping in the polar case

In the case of matching in a polar way, as described in Section 2.2, the mapping $\mathcal{D}_{r2v}^{(\gamma)}(t,s)$ can be defined as

$$\mathcal{D}_{r2v}^{(\gamma)}(t,s) = \begin{bmatrix} \angle(s-p^*) - \omega(t) \\ \|s - \gamma(\angle(s-p^*))\| \end{bmatrix}, \tag{9}$$

(see Fig. 2(b) for a graphical representation of the proposed diffeomorphism). Given a position $s(t)$ in the real frame $\mathcal{F}_r$ then, according to (9) and Proposition 2, it follows that

$$\frac{\partial \mathcal{D}_{r2v}^{(\gamma)}(t,s)}{\partial s} = \begin{bmatrix} \frac{(s-p^*)^T}{\|s-p^*\|^2}A \\ \frac{(s-\gamma(\angle(s-p^*)))^T}{\|s-\gamma(\angle(s-p^*))\|}\left(I_2 - \frac{\partial \gamma}{\omega}\Big|_{\omega(t)}\frac{(s-p^*)^T}{\|s-p^*\|^2}A\right) \end{bmatrix},$$

where I_2 is the identity matrix of order 2, and

$$\frac{\partial \mathcal{D}_{r2v}^{(\gamma)}(t,s)}{\partial t} = \begin{bmatrix} -\dot{\omega}(t) \\ 0 \end{bmatrix}.$$

As in the previous case, in the virtual frame $\mathcal{F}_v$, a control law can be designed by decoupling its action along the two virtual axes. In particular, on the first axis, the agents will be driven so that to be close to the reference profile $\omega(t)$ and to obtain the matching. On the second virtual axis the control action must drive the agents' components to zero so that to reach the desired curve.

4. Model properties in the virtual reference frame

This section is devoted to the analysis of agents moving in the virtual reference frame. In particular, the control actions $\overline{u}_{x,i}(t)$ and $\underline{u}_{x,i}(t)$, $i = 1,\dots,n$ in (6) are designed to solve MP. Using the procedure described in the previous section, all the properties derived in $\mathcal{F}_v$ are then converted into analogous properties in the real frame $\mathcal{F}_r$. The analysis is here conducted considering the curve γ and the generic multi-agent system

$$\dot{x}_i(t) = u_{x,i}(t), \quad i = 1,\dots,n, \tag{10}$$

where $x_i(t) = \overline{x}_i(t)$ (resp. $\underline{x}_i(t)$), $u_{x,i}(t) = \overline{u}_{x,i}(t)$ (resp. $\underline{u}_{x,i}(t)$) and $\gamma = \overline{\gamma}$ (resp. $\gamma = \underline{\gamma}$), respectively. Specifically, $x_i(t)$ represents the transformation from the real frame $\mathcal{F}_r$ to the virtual frame $\mathcal{F}_v$ at the current time t.

Consider the two components of the vector $x_i(t)$, namely $x_i(t) = \begin{bmatrix} \xi_i(t), & \zeta_i(t) \end{bmatrix}^T$ and assume the following evolution

$$\dot{\xi}_i(t) = -\rho\xi_i(t) - \alpha\sum_{j=1}^{n}(\xi_i(t) - \xi_j(t)) + \beta\sum_{\substack{j=1,\\ j\neq i}}^{n}\frac{1}{\xi_i(t)-\xi_j(t)}, \tag{11}$$

$$\dot{\zeta}_i(t) = -\rho\ \mathrm{sign}(\zeta_i(t))|\zeta_i(t)|^{\mu}, \quad i = 1,\dots,n, \tag{12}$$

where $\rho, \alpha, \beta \in \mathbb{R}_{>0}$, $\mu \in (0,1)$ are control parameters. Specifically,

- ρ modulates the attraction to the desired profile;
- α and β are responsible for achieving a balance between the effect of attraction and repulsion with other members of the swarm;
- the parameter μ together with ρ modulates the convergence of $\zeta_i(t)$ in (12) towards zero (as shown in the following proposition).

As a first result, the asymptotic behavior of the virtual signals ξ_i and ζ_i is investigated. In particular, the dynamic (12) guarantees that the two swarms reach the lines $\overline{l}$ and $\underline{l}$, in the tangent mapping case, or the curves $\overline{\gamma}$ and $\underline{\gamma}$ in the polar mapping case, in finite-time.

Proposition 3. *The dynamic (12) tends to zero in the finite-time, i.e.,* $\exists \tau_{\zeta_i} > 0,\ s.t.\ \zeta_i(t) = 0,\ \forall t \geq \tau_{\zeta_i}$.

Proof. To prove the convergence to zero in finite-time, consider the candidate Lyapunov function $V_{\zeta_i}(t) = \frac{1}{2}\zeta_i(t)^2$, whose derivative is

$$\dot{V}_{\zeta_i} = \zeta_i\dot{\zeta}_i = -\rho|\zeta_i|^{\mu+1}$$
$$= -\rho\left(\zeta_i^2\right)^{\frac{\mu+1}{2}} = -\rho 2^{\frac{\mu+1}{2}}\left(\frac{\zeta_i^2}{2}\right)^{\frac{\mu+1}{2}} = -\rho 2^{\frac{\mu+1}{2}}V_{\zeta_i}^{\frac{\mu+1}{2}}.$$

As shown in Fedele, D'Alfonso, and D'Aquila (2018), finite-time convergence to zero after

$$\tau_{\zeta_i} = \frac{V_{\zeta_i}^{1-\eta}(0)}{\rho(1-\eta)2^{\eta}}, \tag{13}$$

with $\eta = \frac{\mu+1}{2}$, occurs. □

The above result guarantees that, after a finite-time

$$\tau_{\zeta} = \max_{i=1,\dots,n}\tau_{\zeta_i}, \tag{14}$$

the dynamics of the swarm in the virtual reference frame $\mathcal{F}_v$ depends only on the first component $\xi_i(t)$, $i = 1,\dots,n$. In the subsequent analysis it is then assumed that $t \geq \tau_{\zeta}$ and it is shown that centroid evolution in the virtual reference frame tends to the origin of $\mathcal{F}_v$.

Proposition 4. *The virtual signals $\xi_i(t)$, $i = 1,\dots,n$, evolving according to (11), asymptotically reach a configuration where the centroid tends to the origin, i.e., the agents organize so that their centroid tends to the chosen motion profile related to $\omega(t)$.*

Proof. The reciprocity of interactions in (11) guarantees that $\sum_{i=1}^{n}\dot{\xi}_i = -\rho\sum_{i=1}^{n}\xi_i$. Therefore, the centroid of ξ_is tends to 0 with a velocity related to ρ. □

The next result proves that the discrepancy between the agent's position and the motion profile is bounded, i.e., agents asymptotically enter a bounded region around the centroid, the size of which is a function of ρ and β and can thus be adjusted by an appropriate choice of control parameters.

Proposition 5. *The agents that evolve according to (11) and (12) move towards and remain within a region around the motion profile bounded by*

$$\mathcal{B}_\xi = \sqrt{\frac{n(n-1)\beta}{2\rho}}. \tag{15}$$

Proof. Let $V_\xi = \frac{1}{2}\sum_{i=1}^n \xi_i^2$ be the candidate Lyapunov function. Its derivative is

$$\begin{aligned}\dot V_\xi &= \sum_{i=1}^n \xi_i\dot\xi_i \\ &= -\rho\sum_{i=1}^n \xi_i^2 - \alpha\sum_{i=1}^n\sum_{\substack{j=1,\\ j\neq i}}^n \xi_i(\xi_i-\xi_j) + \beta\sum_{i=1}^n\sum_{\substack{j=1,\\ j\neq i}}^n \frac{\xi_i(\xi_i-\xi_j)}{(\xi_i-\xi_j)^2} \\ &\le -\rho\xi_i^2 + \beta\frac{n(n-1)}{2}, \quad \forall i.\end{aligned}$$

Then $\dot V_\xi < 0$ if $\xi_i^2 > \frac{n(n-1)\beta}{2\rho}$ which states that each agent enters the region around the centroid with bound $\mathcal{B}_\xi$. □

Remark 6. Note that this bound is conservative and the actual swarm size could be smaller than the obtained bounds. However, this is an indication of how the control parameters β and ρ should be chosen so that the position of the agents is closer to the given profile $\gamma(\omega)$.

The interactions amongst each agent and the related trajectory characterized by the motion profile $\omega(t)$, in terms of attraction and repulsion effects as in (11), can be also investigated by means of the function $J_\xi : \mathbb{R}^n \to \mathbb{R}$. Indeed, by considering that the first component of the agent dynamics in Eq. (11) evolves as the antigradient of

$$J_{\xi_i} = \frac{1}{2}\rho\xi_i^2 + \frac{1}{2}\alpha\sum_{\substack{j=1,\\ j\neq i}}^n (\xi_i-\xi_j)^2 - \frac{1}{2}\beta\sum_{\substack{j=1,\\ j\neq i}}^n \log\left((\xi_i-\xi_j)^2\right), \tag{16}$$

then a novel function is selected as follows:

$$\begin{aligned}J_\xi(\xi) &= \sum_{i=1}^n\left(\frac{1}{2}\rho\xi_i^2 + J_{\xi_i}\right) \\ &= \rho\sum_{i=1}^n \xi_i^2 + \frac{1}{2}\alpha\sum_{i=1}^n\sum_{\substack{j=1,\\ j\neq i}}^n (\xi_i-\xi_j)^2 \\ &\quad - \frac{1}{2}\beta\sum_{i=1}^n\sum_{\substack{j=1,\\ j\neq i}}^n \log\left((\xi_i-\xi_j)^2\right),\end{aligned} \tag{17}$$

where $\xi = [\xi_1, \dots, \xi_n]^T$ is the vector of concatenated states of all the swarm agents. Noting that $J_\xi(\xi)$ is bounded from below, then there exists $J_{\underline\xi}$ such that $|J_{\underline\xi}| < \infty$ and $J_\xi(\xi) - J_{\underline\xi} \ge 0$, thus, (17) can be intended as a Lyapunov function.

Next theorem shows that, at steady-state, the swarm converges to a static configuration.

Theorem 7. *Given the multi-agent system described by (11) and the Lyapunov function (17), then* $\lim_{t\to\infty}\dot\xi_i(t) = 0,\ i = 1, \dots, n$.

Proof. Let $\mathcal{D} = \{\xi \in \mathbb{R}^n \mid \xi_k \neq \xi_r, \forall k \neq r,\ k, r = 1, \dots, n\}$ and $\forall c \in \mathbb{R}_{>0}$, $\Omega_c = \left\{\xi \in \mathcal{D} | J_\xi(\xi) < c\right\}$ be the level sets of $J_\xi(\xi)$. Noting that

$$\frac{\partial J_{\xi_i}}{\partial \xi_i} = -\dot\xi_i, \quad \frac{\partial J_{\xi_i}}{\partial \xi_j} = -\alpha(\xi_i-\xi_j) + \beta\frac{1}{\xi_i-\xi_j},$$

then

$$\begin{aligned}\dot J_\xi(\xi) &= -\sum_{i=1}^n \dot\xi_i^2 + \rho\sum_{i=1}^n \xi_i\dot\xi_i \\ &\quad + \sum_{i=1}^n\sum_{\substack{j=1,\\ j\neq i}}^n \left(-\alpha(\xi_i-\xi_j) + \beta\frac{1}{\xi_i-\xi_j}\right)\dot\xi_j.\end{aligned} \tag{18}$$

The last term in (18) can be rewritten as

$$\sum_{i=1}^n\sum_{\substack{j=1,\\ j\neq i}}^n \left(\alpha(\xi_i-\xi_j) - \beta\frac{1}{\xi_i-\xi_j}\right)\dot\xi_i = \sum_{i=1}^n \left(-\dot\xi_i - \rho\xi_i\right)\dot\xi_i.$$

Therefore, one has that $\dot J_\xi(\xi) = -2\sum_{i=1}^n \dot\xi_i^2 \le 0$ and the regions Ω_c, $\forall c \in \mathbb{R}_{>0}$ are positively invariant sets. Moreover, Ω_c are compact sets in virtue of the continuity property of $J_\xi(\xi)$, $\forall \xi \in \mathcal{D}$. Therefore, LaSalle's invariance principle (see Khalil, 2002) and the results stated in Proposition 5 ensure that any trajectory $\xi(t)$ with initial conditions in Ω_c, asymptotically belongs to the largest invariant set contained in $\left\{\xi \in \mathcal{D} | \dot J_\xi(\xi) \equiv 0\right\}$. □

Remark 8. The above result also ensures the collision avoidance property among the involved agents within a swarm.

In fact, by continuity of $J_\xi \in \mathbb{R}^n$, it follows that $|\xi_i(t) - \xi_j(t)| > 0$ and, under the same arguments of Zavlanos, Tanner, Jadbabaie, and Pappas (2009), it results that if two agents initially move from different poses, they never collide. Furthermore, collision avoidance between different swarms will be addressed in Section 5.2.

It is worth noting that Theorem 7 ensures that each MAS tends to minimize its cost function $J_\xi(\xi)$. The next theorem represents one of the main results of the paper and shows that agents organize themselves in such a way that their final positions do not depend on their initial conditions. This means that for each agent $\bar s_i$ belonging to the first swarm, there is one and only one agent $\underline s_j$ of the second swarm, so that $\mathcal{D}_{r2v}^{(\overline\gamma,1)}(t, \bar s_i(t)) = \mathcal{D}_{r2v}^{(\underline\gamma,1)}(t, \underline s_j(t))$.

Theorem 9. *The multi-agent system defined by (11) reaches a steady-state configuration that is independent from the initial conditions of the agents.*

Proof. According to Theorem 7, agents reach steady-state positions. This means that $\dot\xi_i = 0,\ \forall i = 1, \dots, n$, from which a set of n equations can be obtained as follows

$$-\rho\xi_i - \alpha(n-1)\xi_i + \alpha\sum_{\substack{j=1,\\ j\neq i}}^n \xi_j + \beta\sum_{\substack{j=1,\\ j\neq i}}^n \frac{1}{\xi_i-\xi_j} = 0,$$

for $i = 1, \dots, n$. Using Proposition 4 the third term of the above equation is equal to $-\alpha\xi_i$ and the set of equations is simplified as

$$-\frac{\rho+\alpha n}{\beta}\xi + \Gamma(\xi) = 0, \tag{19}$$

where

$$\Gamma(\xi) = \left[\sum_{\substack{j=1,\\ j\neq 1}}^n \frac{1}{\xi_1-\xi_j}, \sum_{\substack{j=1,\\ j\neq 2}}^n \frac{1}{\xi_2-\xi_j}, \dots, \sum_{\substack{j=1,\\ j\neq n}}^n \frac{1}{\xi_n-\xi_j}\right]^T.$$

Let V be the $n \times n$ Vandermonde matrix defined on the set of the distinct nodes ξ_i, $i = 1, \dots, n$ (Eisinberg & Fedele, 2005) (note that the assumption of distinct nodes stems from the swarm's collision avoidance property), i.e. $V(i,j) = \xi_j^{i-1}$, $i, j = 1, \dots, n$, and consider the identities (Eisinberg & Fedele, 2006)

$$V\xi = \begin{bmatrix} \sum_{k=1}^{n} \xi_k \\ \sum_{k=1}^{n} \xi_k^2 \\ \vdots \\ \sum_{k=1}^{n} \xi_k^n \end{bmatrix}, \quad VF(\xi) = \begin{bmatrix} \sum_{k=1}^{n} \sum_{\substack{p=1,\\ p\neq k}}^{n} \frac{1}{\xi_k - \xi_p} \\ \sum_{k=1}^{n} \sum_{\substack{p=1,\\ p\neq k}}^{n} \frac{\xi_k}{\xi_k - \xi_p} \\ \vdots \\ \sum_{k=1}^{n} \sum_{\substack{p=1,\\ p\neq k}}^{n} \frac{\xi_k^{n-1}}{\xi_k - \xi_p} \end{bmatrix}.$$

Due to the complete connectivity, the ith element of the vector $VF(\xi)$, namely $(VF(\xi))_i$, is (see Mesbahi & Egerstedt, 2010)

$$(VF(\xi))_i = \sum_{k=1}^{n-1} \sum_{p=k+1}^{n} \frac{\xi_k^{i-1} - \xi_p^{i-1}}{\xi_k - \xi_p},$$

where the term $\xi_k^{i-1} - \xi_p^{i-1}$ can be expanded as

$$\xi_k^{i-1} - \xi_p^{i-1} = (\xi_k - \xi_p) \sum_{q=1}^{i-1} \xi_k^{i-q-1} \xi_p^{q-1},$$

thus obtaining

$$(VF(\xi))_i = \sum_{k=1}^{n-1} \sum_{p=k+1}^{n} \sum_{q=1}^{i-1} \xi_k^{i-q-1} \xi_p^{q-1}$$

and

$$-\frac{\rho + \alpha n}{\beta} V\xi + VF(\xi) = 0. \tag{20}$$

It follows that the ith equation of (20) is a multivariate polynomial equation of order i. As stated in the theorem of Bézout in Li (1997), this system then has $n!$ real solutions, i.e. $\xi_k^* \in \mathbb{R}^n$, $k = 1, \ldots, n!$ such that $-\frac{\rho+\alpha n}{\beta} V\xi_k^* + VF(\xi_k^*) = 0, \forall k$. The Theorem 7 states that the agents reach a stationary configuration which is the solution of (20). Moreover, due to the structure of the system (19), any permutation of the above solution still satisfies (19). As a consequence, the $n!$ possible solutions of (20) are all permutations of the stationary configuration reached by the swarm. Depending on their initial conditions, the agents will then organize themselves into one of the possible solutions ξ_k^*, but since all solutions are permutations of the same configuration, the proof follows. □

The complete strategy is then defined in Algorithm 1 where

$$s^*(t) = \begin{cases} \gamma(t) & \text{in the tangent mapping case,} \\ p^* & \text{in the polar mapping case} \end{cases}$$

and the preliminary parameters' setting is encoded at line 0.

Note that in line 4 of the Algorithm 1, the diffeomorphism is defined by Eq. (7) or Eq. (9), in the tangent and in the polar mapping case, respectively. The transformation only requires $s_i(t) - s^*(t)$ instead of $s_i(t)$ and it can then be obtained using information local to the agent, i.e., the displacement w.r.t. a point of interest $s^*(t)$ (related to $\gamma(t)$) and the displacement w.r.t. the other agents.

The previous results suggest precious information about the agents configuration in the two proposed mapping cases. In the tangent reference frame mapping, the agents of the two swarms organize themselves on the two lines $\overline{l}(\psi)$ and $\underline{l}(\psi)$ in such a way that the deviations between their positions and the desired profile $\overline{\gamma}(\omega(t))$ and $\underline{\gamma}(\omega(t))$ respectively are constant and do not depend on the initial conditions of the agents. Moreover, the result given by Proposition 5 ensures that the agents will reach a bounded region around the reference profiles $\overline{\gamma}(\omega(t))$ and $\underline{\gamma}(\omega(t))$. Following the same lines, if a polar mapping is exploited, the agents will reach a configuration on $\overline{\gamma}(\omega(t))$ and $\underline{\gamma}(\omega(t))$ where the deviations between their angles, w.r.t. the vantage point p^*, and the desired motion $\omega(t)$ are constant and do not depend on the initial conditions of the agents too. In this case a bound around the reference profiles is given by

$$\Gamma = \max\{\overline{\Gamma}, \underline{\Gamma}\}, \tag{21}$$

where

$$\overline{\Gamma} = \max_t \left\{ \max_{w \in [\omega(t) - B_\xi, \omega(t) + B_\xi]} \|\overline{\gamma}(w) - \overline{\gamma}(\omega(t))\| \right\},$$

$$\underline{\Gamma} = \max_t \left\{ \max_{w \in [\omega(t) - B_\xi, \omega(t) + B_\xi]} \|\underline{\gamma}(w) - \underline{\gamma}(\omega(t))\| \right\}.$$

Thus, we can conclude that the two swarms reach a marriage condition where each pair of agents moves in a coupled manner along the chosen profile. It is worth noting that no assumption was made about the coupling between the two swarms and that the only assumption is the complete connectivity within each individual swarm. This assumption was necessary in Theorem 9 to ensure that all possible solutions are permutations of a single solution.

Algorithm 1 Control strategy for i-th agent.

0: **Parameters Setting:**
▷ Select $\overline{\alpha}$ and $\underline{\alpha}$ so that the target curves $\overline{\gamma}$ and $\underline{\gamma}$ are at a desired distance from each other.
▷ Select the parameters ρ and μ according to Eqs. (13) and (14) so that the convergence time of the second virtual component of each agent is below a desired value. The more ρ increases and the more μ decreases, the faster the convergence will be.
▷ Select the parameter α considering that as α increases and all other parameters being equal, at steady-state the agents tend to get closer.
▷ Select the parameter β according to Eqs. (15) and (21) so that the boundary of the region in which the agents converge meets the given requirements. The more β grows, the larger the boundary will be.

1: **Input:** $s_i(t) - s^*(t)\ s_i(t) - s_j(t),\ j \neq i,\ \gamma,\ t,\ \mathcal{D}_{r2v}$

2: **Output:** $u_{s,i}(t)$ to be used in Eq. (1)
▷ **Step 1: obtain** $s_j(t) - s^*(t),\ j \neq i$

3: $s_j(t) - s^* = (s_i(t) - s^*(t)) - (s_i(t) - s_j(t))$
▷ **Step 2: transform** $s_i(t) - s^*(t)$ **from** $\mathcal{F}_r$ **to** $\mathcal{F}_v$, $i = 1, \ldots, n$

4: Compute $x_i(t),\ i = 1, \ldots, n$ using $\mathcal{D}_{r2v}^{\gamma}$ as defined in Eq. (7) in the tangent mapping case, or in Eq. (9) in the polar mapping case
▷ **Step 3: solve the MP in** $\mathcal{F}_v$ **using Eqs. (11), (12)**

5: $u_{x,i} = \begin{bmatrix} -\rho\xi_i(t) - \alpha \sum_{j=1}^{n} (\xi_i(t) - \xi_j(t)) + \beta \sum_{\substack{j=1,\\ j\neq i}}^{n} \frac{1}{\xi_i(t) - \xi_j(t)} \\ -\rho\ \mathrm{sign}(\zeta_i(t))|\zeta_i(t)|^{\mu} \end{bmatrix}$
▷ **Step 4:Transform back the control law from** $\mathcal{F}_v$ **to** $\mathcal{F}_r$ **using Eq. (5)**

6: $\mathcal{J}_i(t) = \left. \frac{\partial \mathcal{D}_{r2v}^{(\gamma)}(t,s)}{\partial s} \right|_{s_i(t),t}$

7: $\sigma_i(t) = \left. \frac{\partial \mathcal{D}_{r2v}^{(\gamma)}(t,s)}{\partial t} \right|_{s_i(t),t}$

8: $u_{s,i} = [\mathcal{J}_i(t)]^{-1} \left(u_{x,i} - \sigma_i(t)\right)$

5. Multiple multi-agent systems

In this section some useful considerations will be provided on how to apply the obtained results to the case of multiple multi-agent systems and on how the properties of the single swarm can ensure further features on all the groups of swarms.

5.1. The case of more than 2 swarms

Although all the proposed results have been studied in the case of two swarms of agents, the proposed strategy can be easily extended to the case of a number of multi-agent systems greater than 2. In particular, let the number of swarms be $N \in \mathbb{N}$ and let each agent be denoted as $s_i^{(k)}(t)$, $i = 1, \dots, \nu^{(k)}$, $k = 1, \dots, N$, where the superscript indicates the multi-agent system to which the agent belongs, while the subscript is the index of the agent into its swarm. As the only assumptions for tailoring the proposed model to the case of $N > 2$:

- each swarm has to be formed by the same number of agents, i.e., $\nu^{(k)} = \nu$, $k = 1, \dots, N$;
- each group is assigned a reference curve $\gamma^{(k)}(\omega) \in \mathcal{C}^2$ that is an upward/downward shifted version or a scaled version of the original $\gamma(\omega)$. In the case of open curves, each $\gamma^{(k)}$ can be chosen as $\gamma^{(k)}(\omega) = [\gamma_x(\omega), \gamma_y(\omega) + \alpha^{(k)}]^T$, with $\alpha^{(k)} \in \mathbb{R}$, while in the case of closed curves $\gamma^{(k)}(\omega) = \alpha^{(k)}(\gamma(\omega) - c_\gamma) + c_\gamma$, with $\alpha^{(k)} \in \mathbb{R}_{>0}$. If a collision avoidance between the multi-agent systems is required, the values of $\alpha^{(k)}$ have to be chosen so that to ensure no intersections between the resulting curves and so that the distance between each couple of curves is at least equal to $2B_\xi$. The last assumption, as prescribed by Proposition 5, ensures that at steady-state the agents belonging to a swarm cannot collide with the reference curve of another multi-agent system.

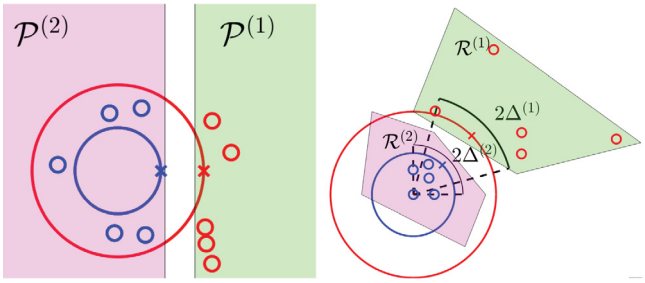
At this point, given the nature of the proposed model, which does not require knowledge or connectivity between different multi-agent systems, but only within the same multi-agent system, the same strategy for the case of $N = 2$ can be easily extended to the case of multiple swarms without requiring changes to the control protocol and maintaining the validity of all the results presented.

5.2. Collision between different swarms

If a single swarm is considered, then Theorem 7 ensures that agents belonging to the same group cannot collide. However, in some practical scenarios, it is also important to analyze the possibility of collision between agents belonging to heterogeneous groups. Since the results presented are all such that a communication is not required between the different swarms, to avoid collisions between heterogeneous multi-agent systems, two solutions can be exploited.

As a first solution, it can be assumed that each agent delegates the absence of clashes to a low-level control module (see Fedele, D'Alfonso, Chiaravalloti & and D'Aquila, 2018; Franze, D'Alfonso, & Fedele, 2018). In this case, the proposed model can be seen as a planner for the trajectories of the real agents that will be driven by properly designed control laws so that to simultaneously ensure trajectory tracking and collision avoidance.

As a second approach, depending on the chosen diffeomorphism $\mathcal{D}_{r2v}^{(\gamma)}$, some considerations on its geometrical meaning can be done. In particular, in the case of the proposed mappings, some simple constraints can be imposed on the initial conditions of the two groups of agents and on the reference motion profile $\omega(t)$. In particular, the results of Propositions 3 and 5 state that, in the virtual reference frame the agents monotonically tend to their reference states. The obtained control law is later transferred to the real world by means of the chosen diffeomorphism and by Proposition 2 and, as a consequence, the convergence properties are held valid (see D'Alfonso, Fedele, & Bono, 2023) and the distance between each agent and its reference curve decreases monotonically. Thus, the collision between different swarms can be avoided by dividing the control protocol in two main steps. In the first step, for $t \in [0, \tau_\zeta]$ the motion profile is keep constant. As a second step, the motion profile can change according to the desired behaviors for the swarms. In the second step, the collisions are for sure avoided if the assumptions in Section 5.1 are satisfied for the curves $\gamma^{(k)}(\omega)$, $k = 1, \dots, N$. In the first control step, i.e., when the motion profile is constant, the two cases of tangent and polar mapping have to be addressed in different ways.



(a) Example of possible half-spaces $\mathcal{P}^{(k)}$, $k = 1, 2$ in the case of a tangent mapping as described in Section 3.1. The green region is related to the first multi-agent system that starts from the red circles and has the red cross as reference point $\gamma^{(1)}(\omega(0))$. The magenta region is related to the second multi-agent system that starts from the blue circles and has the blue cross as reference point $\gamma^{(2)}(\omega(0))$.

(b) Example of possible convex regions $\mathcal{R}^{(k)}$, $k = 1, 2$ in the case of a polar mapping as described in Section 3.2. The green region is related to the first multi-agent system that starts from the red circles and has the red cross as reference point $\gamma^{(1)}(\omega(0))$. The magenta region is related to the second multi-agent system that starts from the blue circles and has the blue cross as reference point $\gamma^{(2)}(\omega(0))$.

Fig. 3. Half-spaces and convex regions considered in the tangent and polar mappings.

5.2.1. Tangent mapping case

If a tangent mapping is taken into account, to avoid collision between multi-agent systems, the following conditions have to be exploited. More precisely, if a half-space $\mathcal{P}^{(k)}$ can be defined for each curve $\gamma^{(k)}(\omega)$, $k = 1, \dots, N$ such that

- $\gamma^{(k)}(\omega(0)) \in \mathcal{P}^{(k)}$;
- $s_i^{(k)}(0) \in \mathcal{P}^{(k)}$, $i = 1, \dots, \nu$;
- the tangent line to $\gamma^{(k)}(\omega(0))$, centered in $\gamma^{(k)}(\omega(0))$, belongs to $\mathcal{P}^{(k)}$;
- $\mathcal{P}^{(i)} \bigcap \mathcal{P}^{(j)} = \varnothing$, $\forall\, i \neq j$;

then a possible overlap of the trajectories of the agents can be excluded from the beginning. An example of possible half-spaces $\mathcal{P}^{(k)}$ in the case of $k = 1, 2$ is depicted in Fig. 3(a).

5.2.2. Polar mapping case

In the case of a polar mapping as described in Section 3.2, collisions between multi-agent systems can be avoided if a convex region $\mathcal{R}^{(k)}$ can be defined for each curve $\gamma^{(k)}(\omega)$, $k = 1, \dots, N$ such that

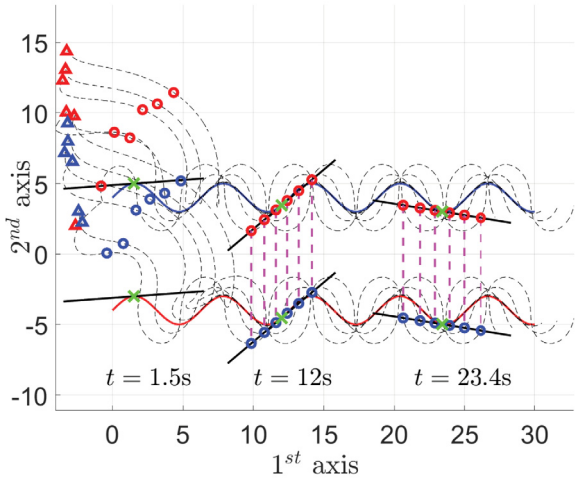


Fig. 4. Example 1: the dotted lines represent the agents' evolution. The first swarm is encoded with red triangles (initial conditions) and circles while the second one is encoded with blue triangles and circles. $\overline{\gamma}$ is the blue line and $\underline{\gamma}$ is the red line. Three time snapshots are highlighted and for each snapshot the green crosses represent the reference points $\overline{\gamma}(\omega(t))$ and $\underline{\gamma}(\omega(t))$ while the related lines $\bar{l}(\psi)$ and $\underline{l}(\psi)$ are the black tangent lines. The magenta dotted lines show the marriage condition once it is achieved. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

- $\gamma^{(k)}(w) \in \mathcal{R}^{(k)}, \forall w \in \left[\omega(0) - \Delta^{(k)}, \omega(0) + \Delta^{(k)}\right]$ with $\Delta^{(k)} = \max\left\{\max_{i=1,\dots,\nu}\left|\xi_i^{(k)}(0)\right|, \mathcal{B}_\xi\right\}$ and $\xi_i^{(k)}(0)$ is the first component of the agent $s_i^{(k)}(0)$ in the virtual reference frame;
- $s_i^{(k)}(0) \in \mathcal{R}^{(k)}, i = 1, \dots, \nu$;
- $\mathcal{R}^{(i)} \bigcap \mathcal{R}^{(j)} = \varnothing, \forall\, i \neq j$.

In this case, as in the previous one, an overlap of the trajectories of the agents can be excluded. An example of possible regions $\mathcal{R}^{(k)}$ in the case of $k = 1, 2$ is depicted in Fig. 3(b).

6. Numerical and experimental results

To further illustrate the obtained results, two numerical simulations and a laboratory experiment have been carried out considering both the cases of tangent and polar diffeomorphisms. The aim of this Section is to clarify the theoretical properties and their benefits in the design of multi-agent systems with matching conditions. In all the considered numerical scenarios, the model parameters have been set to $\alpha = 1.0$, $\rho = 1.0$, $\mu = 0.4$ while the experimental setup will be described in the related section.

6.1. Example 1: Tangent mapping on an open curve

In the first example, the tangent mapping shown in Section 3.1 has been exploited considering 2 swarms of $n = 6$ agents. The repulsion among the agents has been weighted by $\beta = 10$ and as the reference curve is concerned, $\gamma(\omega) = [\omega, \sin(\omega)]^T$ and the motion profile $\omega(t) = t$ have been considered. The values of $\overline{\alpha} = \underline{\alpha} = 4$ have been used to obtain $\overline{\gamma}$ and $\underline{\gamma}$, respectively. The obtained results are shown in Figs. 4 and 5. The first figure shows the time evolution of the two multi-agent systems. It can be noticed how the two groups organize themselves so that to satisfy the condition of the MP as described in Section 2.1, as highlighted by the magenta lines that connects the couple of agents that match. As expected by the shown theoretical results, the agents reach a steady-state condition on the tangent lines $\bar{l}(\psi)$ and $\underline{l}(\psi)$, regardless of their initial conditions. As the achievement of the tangent lines is concerned, Fig. 5 shows the finite-time convergence of the multi-agent systems. In particular, considering a point $P = [P_x, P_y]^T \in \mathbb{R}^2$ and a line defined by the equation $l_a x + l_b y + l_c = 0$, with $l_a, l_b, l_c \in \mathbb{R}$, in the plane defined by axes x and y, the point to line distance $d_{p2l} = \dfrac{\left|l_a P_x + l_b P_y + l_c\right|}{\sqrt{l_a^2 + l_b^2}}$ as described in Spain (2007) has been computed and depicted for each agent in the two swarms. The obtained results in terms of agents state and distance from the computed tangent lines is depicted over time in Fig. 5. As it can be noted, all the agents, in both the swarms, reach and remain on the tangent lines in finite-time before the value τ_ζ, computed according to Proposition 3 and considering the agents' initial conditions.

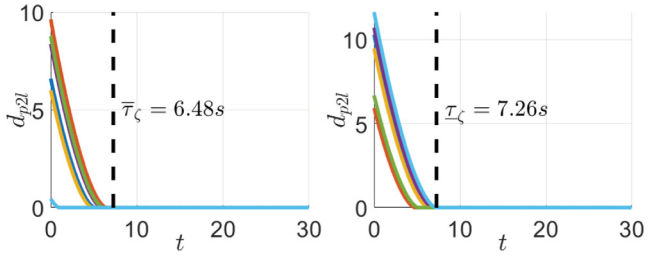


Fig. 5. Example 1: Point to tangent line distances for each agent in the first swarm (on the left, w.r.t. $\bar{l}(\psi)$ with $\tau_\zeta = \overline{\tau}_\zeta$) and in the second swarm (on the right w.r.t. $\underline{l}(\psi)$) with $\tau_\zeta = \underline{\tau}_\zeta$.

6.2. Example 2: Polar mapping

As a second example, the case of polar mapping, as shown in Section 3.2 has been considered with 2 swarms of $n = 4$ agents. The repulsion among the agents has been weighted by $\beta = 2$ and the rhodonea curve, as described in Cundy (1972),

$$\gamma(\omega) = \begin{cases} 7(3 + \cos(5\omega)\cos(\omega)) \\ 7(3 + \cos(5\omega)\sin(\omega)) \end{cases}$$

has been considered as reference curve, with a motion profile $\omega(t) = 8\sin(0.1t)$. The values of $\overline{\alpha} = 1.4$ and $\underline{\alpha} = 0.58$ have been used to obtain $\overline{\gamma}$ and $\underline{\gamma}$, respectively. The obtained results are shown in Fig. 6 in terms of four time snapshots of the agents' evolution. The two swarms satisfy the conditions prescribed by the MP in Section 2.2 in this case too. The magenta lines highlight the couple of agents that are in a matching state. In this scenario as well, the independence of each agent's steady-state condition from its initial condition is ensured.

Considering now the statements in Section 5.1, a further example in the same scenario of Example 3 but with 4 groups of $\nu = 3$ agents, each one with a curve scaled as $\alpha^{(1)} = 20$, $\alpha^{(2)} = 15$, $\alpha^{(3)} = 10$, $\alpha^{(4)} = 5$, is analyzed in Fig. 7. Also in this case the marriage condition is achieved and no changes have been applied to the described model despite the use of more than 2 multi-agent systems.

6.3. Example 3: Matching in a real experiment

A laboratory experiment assessed the proposed algorithm's performance using six Elisa-3 autonomous robots by GCTronic (2023). These circular robots, measuring 50 mm in diameter, 30 mm in height, and weighing 39 g, utilize wireless communication via RF 2.4 GHz. Equipped with wheels connected to a DC motor and a reduction gear of 25 : 1, the robots move within a 0.8 m × 0.6 m arena. A Trust Spotlight PRO RGB camera tracks

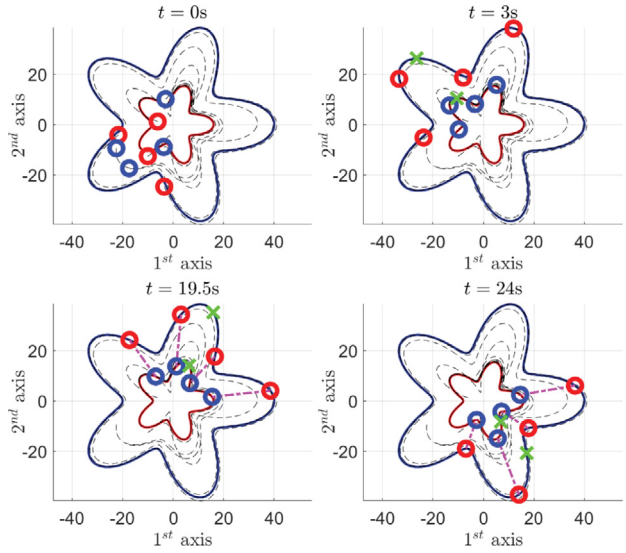


Fig. 6. Example 2: the dotted lines represent the agents' evolution. The first swarm is encoded with red circles while the second one is encoded with blue circles. $\overline{\gamma}$ is the blue line and $\underline{\gamma}$ is the red line. Four time snapshots are highlighted; the green crosses represent the reference points $\overline{\gamma}(\omega(t))$ and $\underline{\gamma}(\omega(t))$. The magenta dotted lines show the marriage condition once it is achieved. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

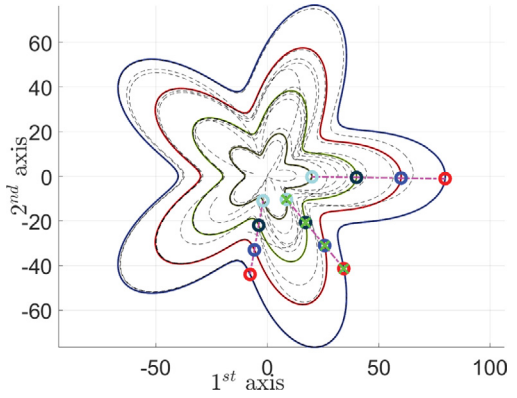


Fig. 7. Multiple swarms example: the dotted lines represent the agents' evolution. All the agents that belong to the same swarm are encoded by the circles of the same color. $\gamma^{(k)}, k = 1, \ldots, 4$ are shown with blue, red, green and black curves, respectively. The steady-state configuration is highlighted and the green crosses represent the reference points for each swarm. The magenta dotted lines show the marriage condition. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

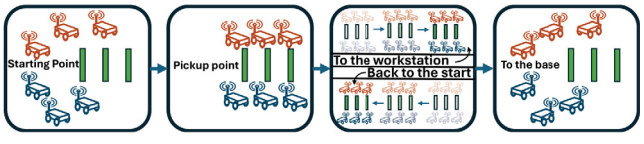


Fig. 8. Example 3: Intelligent transportation problem. The agents (red group and blue group) agree on positions to take loads (green rectangles) and to move to the workstation. The groups then come back to the starting positions to leave the loads and the agents return to their baseline condition. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

their positions, with a video processing module detecting the robot centers. Communication occurs through a radio base station connected to a PC via USB. Each robot evolves according to the well-known kinematic model (see Fedele, D'Alfonso, Chiaravalloti & and D'Aquila, 2018)

$$\begin{cases} \dot{x}_i^R(t) &= v_i(t)\cos\left(\theta_i^R(t)\right) \\ \dot{y}_i^R(t) &= v_i(t)\sin\left(\theta_i^R(t)\right) \\ \dot{\theta}_i^R(t) &= \phi_i(t) \end{cases}, \quad i = 1, \ldots, 6,$$

where x_i^R, y_i^R and θ_i^R defines the position and heading of each robot while v_i and ϕ_i account for the linear and the angular velocities, respectively. As described by Siciliano, Sciavicco, Villani, and Oriolo (2010), this model can be controlled by means of the input–output linearization-based control law

$$\begin{bmatrix} v_i(t) \\ \phi_i(t) \end{bmatrix} = \begin{bmatrix} \cos\left(\theta_i^R(t)\right) & -d\sin\left(\theta_i^R(t)\right) \\ \sin\left(\theta_i^R(t)\right) & d\cos\left(\theta_i^R(t)\right) \end{bmatrix}^{-1} \begin{bmatrix} \tau_{x,i}(t) \\ \tau_{y,i}(t) \end{bmatrix},$$

so that to move the point

$$P_i(t) = \begin{bmatrix} P_{x,i}(t) \\ P_{y,i}(t) \end{bmatrix} = \begin{bmatrix} x_i^R(t) + d\cos\left(\theta_i^R(t)\right) \\ y_i^R(t) + d\sin\left(\theta_i^R(t)\right) \end{bmatrix},$$

with $d \neq 0$, according to $\dot{P}_i(t) = \left[\tau_{x,i}(t),\ \tau_{y,i}(t)\right]^T$. The six robots are divided into two subgroups, SW_1 formed by the first three agents (i.e., $i = 1, 2, 3$) and SW_2 was consisting of the last three ones ($i = 4, 5, 6$). The aim is to emulate a typical intelligent transportation scenario where two groups of vehicles must transport loads by positioning themselves in pairs at their respective ends. Subsequently, the loads are conveyed to a workstation and then returned to the starting point, as depicted in Fig. 8. This scenario can be effectively addressed through the proposed matching solution. Specifically, the path that agents must traverse with the load represents the curve γ, from which $\overline{\gamma}$ and $\underline{\gamma}$ can be defined to ensure that they facilitate load transport. The motion profile $\omega(t)$ can be defined to ensure that vehicles position themselves first at load pickup points (constant $\omega(t)$ indicating agents move toward fixed positions over time), and then begin moving along the defined path toward the workstation and back to the baseline condition. At this point, agents can return to their base stations once the loads are deposited. Notably, during the load pickup and transport phases, the pairwise transport condition is naturally adhered to by the proposed solution, as it is perfectly compliant with the given matching definition. Furthermore, it should be noted that Theorem 9 ensures that the position to deposit the loads to let the agents pick them up can be calculated in advance through a preliminary simulation phase. The control laws $\tau_{x,i}(t)$, $\tau_{y,i}(t)$ can be straightforwardly computed so that to follow the trajectories defined by the proposed matching strategy that acts then as a planner for the overall control scheme. In the considered laboratory setup, for the sake of simplicity, it has been assumed that the path to be traveled for the agents with the loads is a straight line, i.e. $\gamma(\omega) = [0.25 + \omega, 0.3]^T$ with $\overline{\alpha} = \underline{\alpha} = 0.1$. The motion profile $\omega(t)$ is kept constant at zero until the agents reach the positions of the loads while it moves according to $\omega(t) = \omega_{max}\sin^2\left(\frac{t-T_1}{T_2-T_1}\pi\right)$ so that to go to ω_{max} and coming back in the time interval $[T_1, T_2]$, where ω_{max} defines the required displacement to arrive to the workstation. The control parameters $d = 0.02$ and the matching parameters $\rho = 0.15$, $\alpha = 0.001$, $\beta = 0.001$ and $\mu = 0.9$ have been used. The overall agents' evolution can be appreciated in the video (https://www.youtube.com/watch?v=tbj6W6Gczls) and in Fig. 9 where, for the sake of clarity, pictures of possible loads have been virtually added. The agents move to the pickup points and reach them in 10 s. The two multi-agent systems then reach the workstation at $t \approx 40$ s and move back to the starting points at $t \approx 70$ s; at $t \approx 80$ s the robots are in their base points.

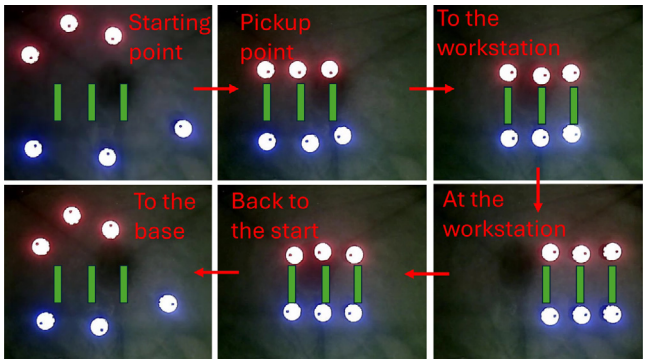


Fig. 9. Example 3: Six snapshots depicting different time instants in the performed experiment. As a first step, the agents match in the pickup positions where they take the loads (virtual loads described by green rectangles) and start their move to the workstation. They then go back to leave the loads and return to the base points. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

7. Conclusions

The paper focused on designing control protocols that enabled two swarms of agents to move along desired paths while achieving pairwise consensus among corresponding agents from each swarm. The proposed solution addressed the matching problem considering the decoupling of the two swarms and the lack of communication between them. This decision to adopt a design that prioritizes privacy and confidentiality ensures that agents belonging to different swarms do not share information and all related confidential information is protected. Future research directions could explore extending these control protocols to three-dimensional spaces, incorporating additional constraints on agent dynamics, and evaluating the proposed approach through extensive simulations or real-world experiments. In addition, methods for dealing with dynamic environments with unknown or changing swarm configurations would be a valuable area for future research.

References

Bono, Antonio, D'Alfonso, Luigi, Fedele, Giuseppe, & Gazi, Veysel (2022). Target capturing in an ellipsoidal region for a swarm of double integrator agents. *IEEE/CAA Journal of Automatica Sinica*, *9*(5), 801–811.

Caccavale, Fabrizio, & Uchiyama, Masaru (2016). Cooperative manipulation. *Springer Handbook of Robotics*, 989–1006.

Cundy, Henry Martyn (1972). *Mathematical models* (2nd ed.). Oxford: Claredon Press.

D'Alfonso, Luigi, Fedele, Giuseppe, & Bono, Antonio (2023). Distributed region following and perimeter surveillance tasks in star-shaped sets. *Systems & Control Letters*, *172*, Article 105437.

Eisinberg, Alfredo, & Fedele, Giuseppe (2005). A property of the elementary symmetric functions. *Calcolo*, *42*(1), 31–36.

Eisinberg, Alfredo, & Fedele, Giuseppe (2006). On the inversion of the vandermonde matrix. *Applied Mathematics and Computation*, *174*(2), 1384–1397.

Farivarnejad, Hamed, Wilson, Sean, & Berman, Spring (2016). Decentralized sliding mode control for autonomous collective transport by multi-robot systems. In *2016 IEEE 55th conference on decision and control* (pp. 1826–1833). IEEE.

Fedele, Giuseppe, D'Alfonso, Luigi, & Bono, Antonio (2023). Emergent hypotrochoidal and epitrochoidal formations in a swarm of agents. *IEEE Transactions on Automatic Control*.

Fedele, Giuseppe, D'Alfonso, Luigi, Bono, Antonio, & Gazi, Veysel (2023). Swarm trajectories generation for target capturing with uncertain information. *IEEE Transactions on Control of Network Systems*.

Fedele, Giuseppe, D'Alfonso, Luigi, Chiaravalloti, Francesco, & D'Aquila, Gaetano (2018). Obstacles avoidance based on switching potential functions. *Journal of Intelligent and Robotic Systems*, *90*, 387–405.

Fedele, Giuseppe, D'Alfonso, Luigi, & D'Aquila, Gaetano (2018). Magnetometer bias finite-time estimation using gyroscope data. *IEEE Transactions on Aerospace and Electronic Systems*, *54*(6), 2926–2936.

Fedele, Giuseppe, D'Alfonso, Luigi, & Gazi, Veysel (2022). A generalized Gazi–Passino model with coordinate-coupling matrices for swarm formation with rotation behavior. *IEEE Transactions on Control of Network Systems*, *9*(3), 1227–1237.

Firoozi, Roya, Zhang, Xiaojing, & Borrelli, Francesco (2021). Formation and reconfiguration of tight multi-lane platoons. *Control Engineering Practice*, *108*, Article 104714.

Fliess, Michel, Lévine, Jean, Martin, Philippe, & Rouchon, Pierre (1995). Flatness and defect of non-linear systems: introductory theory and examples. *International Journal of Control*, *61*(6), 1327–1361.

Franchi, Antonio, Stegagno, Paolo, & Oriolo, Giuseppe (2016). Decentralized multi-robot encirclement of a 3D target with guaranteed collision avoidance. *Autonomous Robots*, *40*, 245–265.

Franze, Giuseppe, D'Alfonso, Luigi, & Fedele, Giuseppe (2018). Distributed model predictive control for constrained multi-agent systems: a swarm aggregation approach. In *American control conference 2018*.

Gale, David, & Shapley, Lloyd S. (1962). College admissions and the stability of marriage. *American Mathematical Monthly*, *69*(1), 9–15.

Gazi, Veysel, & Passino, Kevin M. (2011). *Swarm stability and optimization*. Springer Science & Business Media.

GCTronic (2023). Elisa-3 Documentation. URL http://www.gctronic.com/doc/index.php/Elisa-3.

Gusfield, Dan, & Irving, Robert W. (1989). *The stable marriage problem: Structure and algorithms*. MIT Press.

Johansson, Alexander, Mårtensson, Jonas, Sun, Xiaotong, & Yin, Yafeng (2021). Real-time cross-fleet Pareto-improving truck platoon coordination. In *2021 IEEE international intelligent transportation systems conference* (pp. 996–1003). IEEE.

Khalil, Hassan K. (2002). *Nonlinear systems* (3rd ed.). Upper Saddle River, NJ: Prentice-Hall.

Khamis, Alaa, Hussein, Ahmed, & Elmogy, Ahmed (2015). Multi-robot task allocation: A review of the state-of-the-art. In *Cooperative robots and sensor networks 2015* (pp. 31–51). Springer.

Li, Tien-Yien (1997). Numerical solution of multivariate polynomial systems by homotopy continuation methods. *Acta Numerica*, *6*, 399–436.

Majid, M. H. A., Arshad, M. R., & Mokhtar, R. M. (2022). Swarm robotics behaviors and tasks: a technical review. *Control Engineering in Robotics and Industrial Automation: Malaysian Society for Automatic Control Engineers (MACE) Technical Series 2018*, 99–167.

Meng, Lingxuan, Peng, Zhixing, Zhou, Ji, Zhang, Jirong, Lu, Zhenyu, Baumann, Andreas, et al. (2020). Real-time detection of ground objects based on unmanned aerial vehicle remote sensing with deep learning: Application in excavator detection for pipeline safety. *Remote Sensing*, *12*(1), 182.

Mesbahi, Mehran, & Egerstedt, Magnus (2010). *Graph theoretic methods in multiagent networks*. Princeton University Press.

Papadimitriou, Christos H., & Steiglitz, Kenneth (1998). *Combinatorial optimization: Algorithms and complexity*. Courier Corporation.

Pinto, Miguel, Sobreira, Héber, Moreira, A Paulo, Mendonça, Hélio, & Matos, Aníbal (2013). Self-localisation of indoor mobile robots using multi-hypotheses and a matching algorithm. *Mechatronics*, *23*(6), 727–737.

Plastock, Roy (1974). Homeomorphisms between Banach spaces. *Transactions of the American Mathematical Society*, *200*, 169–183.

Queralta, Jorge Pena, Taipalmaa, Jussi, Pullinen, Bilge Can, Sarker, Victor Kathan, Gia, Tuan Nguyen, Tenhunen, Hannu, et al. (2020). Collaborative multi-robot search and rescue: Planning, coordination, perception, and active vision. *IEEE Access*, *8*, 191617–191643.

Ren, Wei, & Cao, Yongcan (2011). *Distributed coordination of multi-agent networks: Emergent problems, models, and issues*: *vol. 1*, Springer.

Rimon, Elon, & Koditschek, Daniel E. (1991). The construction of analytic diffeomorphisms for exact robot navigation on star worlds. *Transactions of the American Mathematical Society*, *327*(1), 71–116.

Sakurama, Kazunori, Azuma, Shun-Ichi, & Sugie, Toshiharu (2018). Multiagent coordination via distributed pattern matching. *IEEE Transactions on Automatic Control*, *64*(8), 3210–3225.

Siciliano, Bruno, Sciavicco, Lorenzo, Villani, Luigi, & Oriolo, Giuseppe (2010). *Robotics: Modelling, planning and control*. Springer Publishing Company, Incorporated.

Smith, Stephen L., & Bullo, Francesco (2009). Monotonic target assignment for robotic networks. *IEEE Transactions on Automatic Control*, *54*(9), 2042–2057.

Spain, Barry (2007). *Analytical conics*. Courier Corporation.

Tian, Yongding, Chen, Chao, Sagoe-Crentsil, Kwesi, Zhang, Jian, & Duan, Wenhui (2022). Intelligent robotic systems for structural health monitoring: Applications and future trends. *Automation in Construction*, *139*, Article 104273.

Unhelkar, Vaibhav V, Dörr, Stefan, Bubeck, Alexander, Lasota, Przemyslaw A, Perez, Jorge, Siu, Ho Chit, et al. (2018). Mobile robots for moving-floor assembly lines: Design, evaluation, and deployment. *IEEE Robotics & Automation Magazine*, *25*(2), 72–81.

Watanabe, Yuto, & Sakurama, Kazunori (2022). Distributed dynamic matching of two groups of agents with different sensing ranges. In *2022 IEEE 61st conference on decision and control* (pp. 5916–5921). IEEE.

Zavlanos, Michael M, Tanner, Herbert G, Jadbabaie, Ali, & Pappas, George J (2009). Hybrid control for connectivity preserving flocking. *IEEE Transactions on Automatic Control*, *54*(12), 2869–2875.

Zhao, Shiyu, & Sun, Zhiyong (2017). Defend the practicality of single-integrator models in multi-robot coordination control. In *2017 13th IEEE international conference on control & automation* (pp. 666–671). IEEE.

Giuseppe Fedele received the Laurea (M.Sc) degree (cum laude) in computer science engineering and the Ph.D. degree in computer science and system engineering from the University of Calabria (Unical), Rende, Italy, in 1999 and 2005, respectively. Since 2022, he has been an Associate Professor in control engineering with the Department of Informatics, Modeling, Electronics and Systems Engineering (Unical). He is a Founding Member of GiPStech s.r.l., a startup and spin-off of Unical, which designs novel solutions for the indoor positioning and navigation problems. His current research interests include power systems, identification and filtering methods, adaptive control, adaptive algorithms for active noise and vibration control, signal processing for positioning, navigation, and tracking, multi-agent systems. He is the Guest Editor of the Special Issue "Recent Advances in Adaptive Methods for Frequency Estimation with Applications", Int. J. of Adaptive Control and Signal Processing, 2016. He currently serves as an associate editor in Results in Control and Optimization, Franklin Open, European Journal of Control, IEEE Transactions on Cybernetics.

Luigi D'Alfonso was born in Italy in 1985. He received his Ph.D. in Computer Science and Systems Engineering from the University of Calabria, Italy, in 2014. From 2011 to 2014, he was a Ph.D. student at the Department of Computer Science, Modeling, Electronics, and Systems Engineering at the University of Calabria. In 2012, he was a visiting Ph.D. student at the Service d'Automatique et d'Analyse des Systèmes at the Université Libre de Bruxelles, Belgium. From 2015 to 2021, he was a Principal Researcher at GiPStech S.r.l. Since 2022, he has been a Research Fellow at the Department of Computer Science, Modeling, Electronics, and Systems Engineering, University of Calabria, Rende, Italy. His current research interests include mobile robots control, positioning, mapping/SLAM for single and multi-agent systems, the Perspective-n-Point problem using cameras and inertial measurement units, and the modeling and control of swarms of agents.

Boli Chen obtained his B.Eng. in Electrical and Electronic Engineering from Northumbria University, UK, in 2010. He went on to earn an M.Sc. and a Ph.D. in Control Systems from Imperial College London, UK, in 2011 and 2015, respectively. Currently, he is a Lecturer in the Department of Electronic and Electrical Engineering at University College London (UCL), UK. Dr. Chen is a Senior Member of the IEEE and serves as an Associate Editor for the European Journal of Control. Additionally, he is on the EUCA Conference Editorial Board and the editorial board for IEEE ITSC. His research focuses on the control, optimization, and estimation of complex dynamical systems, with applications in the automotive, transportation, and electric energy sectors.