IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 15 June 2022, accepted 27 July 2022, date of publication 16 August 2022, date of current version 14 September 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3198993

RESEARCH ARTICLE

Bimodal Dynamic Swarms

PETER TRAVIS JARDINE[1], (Member, IEEE), AND SIDNEY N. GIVIGI, JR.[2], (Senior Member, IEEE)
[1]Department of Electrical and Computer Engineering, Royal Military College of Canada, Kingston, ON K7K 7B4, Canada
[2]School of Computing, Queen's University, Kington, ON K7K 7B4, Canada

Corresponding author: Sidney N. Givigi, Jr. (sidney.givigi@queensu.ca)

This work was supported in part by the Natural Sciences and Engineering Council of Canada under the Discovery Grant Program.

ABSTRACT This paper presents a bimodal control strategy for transitioning between different topologies of flocking and dynamic structures. The work is motivated by applications where agents must pursue and close in on a target prior to carrying out more sophisticated tasks. Specifically, it builds on previous work in consensus-based flocking to define a new type of dynamic target pursuit and capturing method called dynamic enspherement. The problem is formulated in terms of the canonical Reynolds rules and overlapping planes of encirclement using quaternions of arbitrary orientation. A new method for transitioning between flocking and enspherement is presented, the smoothness of which is demonstrated mathematically. The proposed approach is validated in a series of simulations on swarms of particles with double-integrator dynamics.

INDEX TERMS Decentralized control, bio-inspired control, swarm robotics, drones, quaternions.

I. INTRODUCTION

The field of control is rich with biologically inspired strategies for multiagent coordination and cooperation. The underlying problem of how to analyze and manipulate large groups of distributed agents in computational tractable ways without global knowledge is ubiquitous. Applications for such strategies are far-reaching and include the domains of economics, social science, biology, physics, and robotics. It is no surprise, then, that it is in nature that we find inspiration for the most elegant and efficient algorithms. An enduring theme of multiagent control theory is that of emergent behaviors such as flocking. Whereas formation-based techniques rely on convergence to a desired configuration [1], flocking focuses on convergence of velocities and orientations; the resultant configuration of the flock is an emergent quality. The primary benefits of these emergent behaviour techniques are their scalability and flexibility, due mainly to their distributed nature and reliance on local information. Distributed approaches such as flocking are also more robust than leader-follower-based approaches, since the integrity of the group is generally not impacted by the loss of individual agents [2].

A. RELATED WORK

Early work in distributed multiagent control includes the mechanical approaches in [3], collective motion in [4], and density-based techniques in [5]. The seminal work of Reynolds in 1986 laid the foundation for a large body of research based on heuristic rules of cohesion, alignment, and separation [6]. Many studies have investigated the properties of Reynolds rules of flocking when applied to agents as particles [7] and in the two-dimensional case for reduced complexity [8], [9]. Intuitively, there is an inherent conflict between cohesion (the tendency to move towards other agents in the flock) and separation (the tendency to avoid colliding with other agents in the flock). Therefore, many studies focus on specific subsets of the rules, such as the alignment problem [10], [11].

Contemporary research has aimed at resolving the three-dimensional case by importing distributed optimization-based techniques [12] or consensus problems using network topology and graph theory [13], [14]. An oft-cited theoretical framework for consensus-based approach is found in [15], which formulates a stable, leaderless, lattice-based flocking strategy that avoids obstacles.

Somewhere between formation and flocking methods lies a related body of distributed algorithms that we refer to as *structured swarming*. Like flocking, these are velocity- or alignment-based methods aimed at maintaining a dynamic configuration around a target. Examples include surrounding control [16], cyclic pursuit [17], target capturing [18], circumnavigation [19], and various other dynamic encirclement techniques [20], [21], and [22]. While operating in

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

three-dimensional space, previous encirclement strategies are based on forming a two-dimensional shape around a target. In some cases, it may be desirable to increase coverage of the target by forming a three dimensional structure, such as a dynamic sphere [23].

In certain applications, it may be desirable to switch between these different classes of multiagent techniques. For example, a set of agents may transit towards a target as a flock and then switch to a dynamic encirclement strategy. In such cases, smooth transitions save energy and help preserve stability characteristics between different modes. Some work has investigated the case for time-varying network topologies, such as changing leaders [24], edges [25], or references [26]. In [27], the authors propose a decentralized technique for switching between a fixed formation and dynamic encirclement based on the location of individual agents. In [28], the authors propose a strategy for switching between continuously changing topologies. Similarly, in this work, we investigate the case of continuously switching between multiple discretely defined topologies in a decentralized manner.

B. CONTRIBUTIONS

This work builds on the previous work in flocking [15], structured swarming [22], and time-varying topologies [27] by making the following contributions:

1) reformulation of dynamic encirclement strategies in previous work for application in an arbitrary plane using quaternion-based methods;
2) formulation of a dynamic encirclement algorithm that embodies Reynolds rules of flocking;
3) formulation of a new type of quaternion-based structured swarming method that captures a target within a dynamic sphere (enspherement); and
4) mathematical proof of smoothness for a bimodal control strategy that transitions between flocking and enspherement.

While previous work has developed strategies for producing the individual behaviours above, to the authors' knowledge, this is the first to formulate provably smooth transitions between them.

C. STRUCTURE

The remainder of this paper is structured as follows: Section II presents preliminary information such as notation and models; Section III provides an overview of the flocking technique used; Section IV formulates the bimodal control strategy for transitioning between flocking and enspherement; Section V presents a discussion about smoothness of the proposed technique; Section VI presents experimental results; and Section VII concludes the work.

II. PRELIMINARIES

This section presents the notation used throughout the paper. It also presents the double-integrator model used to define the dynamics of the agents. Finally, a thorough description of the various interactions between agents is provided.

A. NOTATION

The following notation is used throughout this paper: $z(t)$ denotes z is a function of t; $\dot{z}$ is the time rate of change of z; vector z^T is the transpose of z; $||z||$ is the Euclidean norm; $\hat{z}_{i,k}$ is the position of a *virtual* agent derived from the positions of agents i and k; and $\angle(y, x)$ is the angle between vectors y and x.

In order to construct smooth collective potential functions, we use the σ-norm vector $||\cdot||_\sigma$ defined in [15] as follows:

$$||z||_\sigma = \frac{1}{\epsilon}[\sqrt{1+\epsilon||z||^2} - 1] \tag{1}$$

where parameter $\epsilon > 0$. We further define a *bump function* $\rho_h(\cdot)$, which smoothly varies between 0 and 1, as follows:

$$\rho_h(z) = \begin{cases} 1, & z \in [0, h) \\ \frac{1}{2}[1+\cos(\pi\frac{z-h}{1-h})], & z \in [h, 1] \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

where parameter $h \in (0, 1)$.

Finally, in order to find the shortest path to encircle a target within an arbitrary plane, we define the custom $\Pi(\cdot)$ function. Let us define quaternion $\boldsymbol{\rho}$, which will be used to describe the orientation of a desired plane of encirclement. Let us consider any two points q_1 and q_2 in a 3-dimensional Cartesian coordinate system with orthogonal axes x, y, and z. If an agent is rotating in an arbitrary plane defined by $\boldsymbol{\rho}$ at location q_1, we rotate this point about q_2 to orient the plane in the horizontal using the Hamilton product of the inverse quaternion as follows: $\tilde{q}_1 = \boldsymbol{\rho}^{-1}(q_1 - q_2)\boldsymbol{\rho} + q_2$ [29]. We then define a new point, $\Pi(q_1, q_2, r, \boldsymbol{\rho})$, which is the point at distance r from q_2 (i.e. the target of encirclement) in the plane perpendicular to axes z rotated by $\boldsymbol{\rho}$ about q_2:

$$\Pi(q_1, q_2, r, \boldsymbol{\rho}) = \boldsymbol{\rho}\begin{bmatrix} r_\pi(\tilde{q}_{1,x} - q_{2,x}) \\ r_\pi(\tilde{q}_{1,y} - q_{2,y}) \\ 0 \end{bmatrix}\boldsymbol{\rho}^{-1} + \begin{bmatrix} q_{2,x} \\ q_{2,y} \\ q_{2,z} \end{bmatrix} \tag{3}$$

where the norm between $\tilde{q}_1$ and q_2 in the xy-plane $||\tilde{q}_{1,x:y} - q_{2,x:y}||$ is used to define $r_\pi = r/||\tilde{q}_{1,x:y} - q_{2,x:y}||$. In the context of dynamic encirclement, the new point is the closest point on a circle of radius r around target q_2 in the plane of rotation defined by quaternion $\boldsymbol{\rho}$ (i.e. the desired setpoint for an agent at q_1 tasked to encircle the target).

B. AGENT DYNAMICS

We consider all vehicles as dynamic agents with position (q_i), velocity (p_i), and acceleration inputs (u_i) related by the following equations of motion:

$$\begin{aligned} \dot{q}_i &= p_i \\ \dot{p}_i &= u_i \end{aligned} \tag{4}$$

where q_i is expressed in 3-dimensional Cartesian coordinates and $u_i = \ddot{q}_i$ is the acceleration of the agent.

We consider the full network of agents as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with vertices $\mathcal{V} = \{1, 2, \ldots, \eta\}$ and edges $\mathcal{E} \subseteq \{(i,j) : i,j \in \mathcal{V}, j \neq i\}$, where η is the total number of agents. The set of neighbors of the ith agent in space is defined as [15]:

$$N_i = \{j \in \mathcal{V} : a_{ij} \neq 0\} = \{j \in \mathcal{V} : (i,j) \in \mathcal{E}\} \quad (5)$$

where a_{ij} denotes an element of the symmetric adjacency matrix $A[a_{ij}]$ of the graph. In practice, the agents interact with a time-varying subset of neighbours in the space N_i. Depending on the mission and state, the agents may be interacting with any combination of the subsets defined in Section II-C.

C. INTERACTIONS

We describe any *physical* agent with the dynamics described by (4) (i.e. the actual vehicles) as α-agents [15]. Each α-agent interacts with other physical and *virtual* agents in the network, based on relative position. In the formulations that follow, we describe inter-agent interaction with reference to four types of agents:

- α-agents interact for flocking and are contained in the set $\mathcal{V}_\alpha$.
- β-agents interact for collision/obstacle avoidance and are contained in the set $\mathcal{V}_\beta$.
- δ-agents interact for maintaining angular separation in a common plane of dynamic encirclement and are contained the set $\mathcal{V}_\theta$.
- γ-agents interact for tracking a collective objective or target and are contained in the set $\mathcal{V}_\gamma$.

For the purposes of flocking, the ith agent interacts with neighbors $j \in \mathcal{V}_\alpha$ (other α-agents) in order to maintain separation distance $||q_j - q_i|| = d$ as follows:

$$N_i^\alpha = \{j \in \mathcal{V}_\alpha : ||q_j - q_i|| < r\} \quad (6)$$

where $r \geq d > 0$ is the interaction range (typically determined by sensor range). In order to avoid neighboring obstacles, we define β-agents $k \in \mathcal{V}_\beta$ with positions and velocities $(\hat{q}_{i,k}, \hat{p}_{i,k})$ and minimum separation $||\hat{q}_{i,k} - q_i|| = d'$ as follows:

$$N_i^\beta = \{k \in \mathcal{V}_\beta : ||\hat{q}_{i,k} - q_i|| < r'\} \quad (7)$$

where $r \leq r'$ is the interaction range for obstacles. Note that $\hat{q}_{i,k}$ is not the physical location of the obstacle; rather, it is a virtual agent derived from the location and characteristics of the obstacle. The precise definition of this virtual agent will be further defined in Section IV.

Finally, we present a special formulation for dynamic encirclement (which is then extended for enspherement). The goal of dynamic encirclement is for all agents in the formation to form an evenly-spaced circle around the target at a desired angular rate of speed $\dot{\theta}_r$. We define the set of all agents in the encirclement formulation with the ith agent as N_i^θ such that the desired angular separation $\Delta\theta_{d,i}$ is as follows:

$$\Delta\theta_{d,i} = \frac{2\pi}{n_\theta} \; \forall \, i \in \mathcal{V}_\theta \quad (8)$$

where n_θ is the total number of agents in the common plane of dynamic encirclement (i.e. the number of agents in $\mathcal{V}_\theta$). However, agents must form this circle with reference to neighboring agents only (i.e. without global knowledge). Therefore, let us define indices for the subset of local agents in the larger formation ($N_i^\delta \in N_i^\theta$) below. The ith agent encircling target at position q_r regulates its position with reference to the agent immediately *leading* at angle $\theta_i^+ \in [0, 2\pi]$ and agent immediately *lagging* at angle $\theta_i^- \in [0, 2\pi]$:

Let us define the agents leading ($N_i^{\delta+}$) and lagging ($N_i^{\delta-}$) agent i at steady-state as follows:

$$\begin{aligned} N_i^{\delta+} &= \{l \in \mathcal{V}_\theta \; : \angle(q_l, q_r) = \theta_i^+\} \\ N_i^{\delta-} &= \{l \in \mathcal{V}_\theta \; : \angle(q_l, q_r) = \theta_i^-\} \end{aligned}$$

where $\theta_i^+ = \angle(q_i, q_r) + \Delta\theta_{d,i}$ and $\theta_i^- = \angle(q_i, q_r) - \Delta\theta_{d,i}$ and $N_i^\delta = \{N_i^{\delta+}, N_i^{\delta-}\} \in N_i^\theta$. In practice, agent i regulates its angular separation with reference to agents in N_i^δ by minimizing a time-varying error term $e_i(t)$. This is accomplished by tracking an intermediary setpoint for desired angular speed, $\dot{\theta}_{r,i}$, the formulation for which is the topic of Section IV.

III. FLOCKING AS A LATTICE

This section summarizes the main equations for flocking in free space (*free flocking*) used in this paper. Specifically, we use the distributed target tracking and lattice flocking formulated in the seminal work [15]. The goal of this type of flocking is for each α-agent to form a lattice pattern with neighboring α-agents while tracking a target. The lattice involves a gradient-based positioning term and velocity consensus term. Target tracking is achieved via a simple feedback controller. Later, we integrate these equations with the dynamic enspherement in a bimodal control strategy.

In free flocking, each agent i in the flock applies control inputs as follows:

$$u_i = u_i^\alpha + u_i^\gamma \quad (9)$$

where u_i^α forms the lattice and u_i^γ tracks the target. The geometry of the lattice is formed by solutions solving the following constraints:

$$||q_j - q_i|| = d \; \forall \, j \in N_i^\alpha \quad (10)$$

where d is the desired separation between agents in the lattice (also known as the lattice *scale*). Let us further define the elements of the spatial adjacency matrix as follows:

$$a_{ij}(q) = \rho_h\left(\frac{||q_j - q_i||_\sigma}{r_\alpha}\right) \quad \in [0, 1], \; j \neq i \quad (11)$$

where $r_\alpha = ||r||_\sigma$. As described in [15], the rules for lattice flocking are defined as follows:

$$\begin{aligned} u_i^\alpha = {} & c_1^\alpha \sum_{j \in N_i^\alpha} \phi_\alpha(||q_j - q_i||_\sigma)\boldsymbol{n}_{ij} \\ & + c_2^\alpha \sum_{j \in N_i^\alpha} a_{ij}(q)(p_j - p_i) \end{aligned} \quad (12)$$

where

$$\phi_\alpha = \rho_h(z/r_\alpha)\phi(z - d_\alpha) \quad (13)$$

$$\phi(z) = \frac{1}{2}[(a+b)\sigma_1(z+c) + (a-b)] \quad (14)$$

$$\boldsymbol{n}_{ij} = \frac{q_j - q_i}{\sqrt{1+\epsilon||q_j - q_i||^2}} \tag{15}$$

$$\sigma_1(z) = \frac{z}{\sqrt{1+z^2}} \tag{16}$$

and the parameters a, b, and c in (14) are chosen such that $\phi(0) = 0$ and $0 < a \leq b$, $c = a - b/\sqrt{(4ab)}$ [15].

The rule in (12) is insufficient to guarantee flocking behavior, as it is susceptible to regular fragmentation. This is resolved by incorporating a simple feedback control rule as follows:

$$u_i^\gamma = -c_1^\gamma \sigma_1(q_i - q_r) - c_2^\gamma (p_i - p_r) \tag{17}$$

Let us now use the structure of flocking used above as inspiration for a dynamic enspherement strategy.

IV. SWARMING AS A SPHERE

In this section, we reformulate the 2-dimensional *dynamic encirclement* strategy defined in [22] as a decentralized structured swarming algorithm with rules similar to the work of Reynolds [6] and compatible with the framework in [15]. We then extend this to the 3-dimensional case, which we refer to as *enspherement*.

For the ith vehicle in the swarm, the algorithm consists of the following terms:

$$u_i = u_i^\delta + u_i^\zeta + u_i^\beta \tag{18}$$

where u_i^δ is the *angular separation* term, u_i^ζ is the *radial tracking* term, and u_i^β is the *collision avoidance* term. Note that u_i^δ is derived with reference to local information only (i.e. just leading and lagging agents) and u_i^ζ only depends the relative position of the target.

A. ANGULAR SEPARATION TERM

Let us consider an agent at (q_i, p_i) tasked to encircle a target at (q_r, p_r) at a desired rate $\dot{\theta}_r$ in the plane normal to the $\tilde{z}$-axis. We denote axis $\tilde{z}$, as the rules that follow may be implemented for encirclement about any arbitrary axis. Axes $\tilde{x}$, and $\tilde{y}$ denote the remaining orthogonal axes, which define the plane in which encirclement occurs.

We first compute q_i', which is the result of a translation in reference frame that places the target q_r at the origin:

$$q_i' = q_i - q_r \tag{19}$$

Let $\Delta\theta_{ij}$ define the angular difference between the ith and jth agent as follows:

$$\Delta\theta_{ij} = \theta_j - \theta_i \in [0, 2\pi] \tag{20}$$

Recall the desired angular difference between all agents is $\Delta\theta_d = 2\pi/n_\theta$, where n_θ is the total number of agents encircling the target. Given a desired angular speed around target centered at q_r of $\dot{\theta}_r$, we define the a desired angular speed for each vehicle in the $\tilde{x}$-$\tilde{y}$-plane as $\dot{\theta}_{r,i}$. The value of $\dot{\theta}_{r,i}$ differs from $\dot{\theta}_r$ when making correction. The desired *emergent* behavior over time t is as follows:

$$\lim_{t\to\infty} \dot{\theta}_{r,i} = \dot{\theta}_r \ \forall\, i \in N_i^\theta$$

Let us consider n_θ agents traveling in a common plane around the origin such that $\Delta\theta_{ki}(t) = \theta_i(t) - \theta_k(t)$ is the angular difference between agent i and leading agent k at time t, $\Delta\theta_{ij}(t) = \theta_j(t) - \theta_i(t)$ is the angular difference between agent i and lagging agent j, $\Delta\theta_d$ is the desired angular difference between each agent, and $\dot{\theta}_r$ is the desired angular speed of rotation in the plane. From the work in [22], we know that setting the desired angular speed $\dot{\theta}_{r,i}(t)$ as follows:

$$\dot{\theta}_{r,i}(t) = \frac{3\dot{\theta}_r + \gamma(\Delta\theta_{ki}(t) - \Delta\theta_{ij}(t))}{3} \ \forall\, i \in N_i^\theta \tag{21}$$

results in system that is Lyapunov stable with steady state $\dot{\theta}_{r,i}(t) = \dot{\theta}_r(t) \ \forall\, i \in N_i^\theta$ and $\gamma > 0$.

The approach above was inspired by the strategy for cyclic pursuit investigated in [18] and [17], with minor modifications to incorporate feedback from both a leading and lagging agent. In [22], the authors regulate position and velocity using model predictive control (MPC) formulated in the polar frame.

In order to leverage the relationship above for the dynamics in (4), we formulate a new feedback control methodology. Since the dynamics are in the Cartesian frame and (21) is in the polar frame, we begin by defining the relationship between these frames. We introduce an abbreviated transformation $\Phi(\cdot)$ tailored for our application that links angular speed $\dot{\theta}$ within the plane defined by quaternion $\boldsymbol{\rho}$. The plane of rotation is described as that which is normal to axis z after it is rotated by $\boldsymbol{\rho}$ about q_r using the Hamilton product [29]:

$$\Phi(q, \dot{\theta}, \boldsymbol{\rho}) = (\boldsymbol{\rho}(\dot{\theta}\bar{\boldsymbol{u}})\boldsymbol{\rho}^{-1}) \times q \tag{22}$$

where unit vector $\bar{\boldsymbol{u}} = [0\ 0\ 1]^T$. Therefore, in order for agent i to encircle the target with an angular speed $\dot{\theta}_r$ in a plane rotated $\boldsymbol{\rho}_i$ from the horizontal, we define control signal u_i^δ:

$$u_i^\delta = -c_1^\delta (p_i - \Phi(q_i', \dot{\theta}_{r,i}, \boldsymbol{\rho}_i) - p_r) \tag{23}$$

where q_i' is the agent position relative to the target, velocity p_i is the agent velocity, p_r is the target velocity, $\dot{\theta}_{r,i}$ is computed using (21), and c_1^δ is a gain parameter. In short, (23) is the controller that achieves the behavior in (21) for each agent in the network. However, this is insufficient to achieve full dynamic encirclement, as the radius must also be tracked.

B. RADIAL TRACKING TERM

In this section, we complete the encirclement behavior by formulating a term for radial tracking. Recall the function $\Pi(\cdot)$ which, given an point in 3-dimensional space, finds the closest point on a circle of a given radius within an arbitrary plane defined by quaternion rotation $\boldsymbol{\rho}_i$. Given agent i at position q_i tasked to encircle target at position q_r at radius r_r in the plane normal to axis z after it is rotated by $\boldsymbol{\rho}_i$, we compute control signal u_i^ζ:

$$u_i^\zeta = -c_1^\zeta (q_i - \Pi(q_i, q_r, r_r, \boldsymbol{\rho}_i)) \tag{24}$$

where $-c_1^\zeta$ is a gain parameter.

C. COLLISION AVOIDANCE

While the terms in Section IV-A and Section IV-B are sufficient to perform to perform a dynamic encirclement maneuver, they do not address the problem of collision avoidance. The angular separation term in (23) has the effect of ensuring agents in the swarm are separated (and hence partially serving a collision avoidance function) but this does not address the problem of agents outside $\mathcal{V}_\theta$. Further, as will be discussed later, a dedicated collision avoidance term will be useful when we combine multiple decentralized encirclement swarms to achieve enspherement.

In this section, we borrow from the work in [15] to integrate collision avoidance into the dynamic encirclement strategy above. Notation from this previous work has been preserved where possible. Here we introduce a new set of virtual β-agents $\mathcal{V}_\beta$ that are induced whenever an obstacle (such as a neighbouring agent, a solid object in space, on an impermissible region) comes within the region defined by radius $r' > 0$ around the agent. More precisely, the indices for the set of obstacles to be considered N_i^β is as follows:

$$N_i^\beta = \{k \in \mathcal{V}_\beta \ : \ ||\hat{q}_{i,k} - q_i|| < r'\} \tag{25}$$

where $\hat{q}_{i,k}$ is the position of the β-agent induced by the kth obstacle neighbouring agent at position q_i. The constraint to maintain a safe distance between agents and obstacles (i.e. collision avoidance) is therefore defined as:

$$||\hat{q}_{i,k} - q_i|| = d' \ \forall \ k \in N_i^\beta \tag{26}$$

where d' is the desired separation and, generally, $d' < r'$. The full state of the β-agent, $(\hat{q}_{i,k}, \hat{p}_{i,k})$ also depends on the nature of the obstacle.

Let us define the β-agent induced by kth obstacle as having one of two modes: the mode corresponding to a spherical obstacle (centered at position $y_{s,k}$ with radius $r_{s,k}$); or the mode corresponding to a hyperplane boundary (passing through point $y_{h,k}$ with unit normal $\boldsymbol{a}_{h,k}$). We then define the β-agent using one of the following equations:

1) For **Spherical Obstacle** centered at position $y_{s,k}$ with radius $r_{s,k}$, we define the β-agent as follows:

$$\hat{q}_{i,k} = \mu q_i + (1-\mu) y_{s,k} \ , \ \hat{p}_{i,k} = \mu P_s p_i \tag{27}$$

where parameter $\mu = r_{s,k}/||q_i - y_{s,k}||$ and $P_s = I - \boldsymbol{a}_k \boldsymbol{a}_k^T$ is a projection matrix computed using unit vector $\boldsymbol{a}_k = (q_i - y_{s,k})/||q_i - y_{s,k}||$.

2) For **Hyperplane Boundary** passing through point $y_{h,k}$ with unit normal $\boldsymbol{a}_{h,k}$, we define the β-agent as follows:

$$\hat{q}_{i,k} = P_h q_i + (I - P_h) y_{h,k}, \quad \hat{p}_{i,k} = P_h p_i \tag{28}$$

where $P_h = I - \boldsymbol{a}_{h,k} \boldsymbol{a}_{h,k}^T$ is a projection matrix computed using the norm of the hyperplane boundary.

We also define the adjacency between agent i and these neighboring β-agents as follows:

$$b_{i,k}(q) = \rho_h \left(\frac{||\hat{q}_{i,k} - q_i||_\sigma}{d_\beta} \right) \tag{29}$$

where $d_\beta < r_\beta$ given $d_\beta = ||d'||_\sigma$ and $r_\beta = ||r'||_\sigma$. Similar to the potential field approach pioneered by [30], we define a repulsive function $\phi_\beta(z)$ to drive agents around obstacles:

$$\phi_\beta(z) = \rho_h \left(\frac{z}{d_\beta} \right) \sigma_1((z - d_\beta) - 1) \tag{30}$$

where $\sigma_1(z) = z/\sqrt{1 + ||z||^2}$. Finally, we define a flocking rule that achieves collision avoidance as follows:

$$\begin{aligned} u_i^\beta &= c_1^\beta \sum_{j \in N_i^\beta} \phi_\beta(||\hat{q}_{i,k} - q_i||_\sigma) \hat{\boldsymbol{n}}_{i,k} \\ &\quad + c_2^\beta \sum_{j \in N_i^\beta} b_{i,k}(q)(\hat{p}_{i,k} - p_i) \end{aligned} \tag{31}$$

where c_1^β and c_2^β are gain parameters and

$$\hat{\boldsymbol{n}}_{i,k} = \frac{q_j - q_i}{\sqrt{1 + \epsilon ||\hat{q}_{i,k} - q_i||^2}} \tag{32}$$

D. DYNAMIC ENCIRCLEMENT AND ENSPHEREMENT

In summary, we combine the results from Section IV-A, Section IV-B, and Section IV-C to form the flocking rules for collision-free dynamic encirclement:

$$\begin{aligned} u_i^\delta &= -c_1^\delta (p_i - \Phi(q_i', \dot{\theta}_{r,i}, \boldsymbol{\rho}_i) - p_r) \\ u_i^\zeta &= -c_1^\zeta (q_i - \Pi(q_i, q_r, r_r, \boldsymbol{\rho}_i)) \\ u_i^\beta &= c_1^\beta \sum_{j \in N_i^\beta} \phi_\beta(||\hat{q}_{i,k} - q_i||_\sigma) \hat{\boldsymbol{n}}_{i,k} \\ &\quad + c_2^\beta \sum_{j \in N_i^\beta} b_{i,k}(q)(\hat{p}_{i,k} - p_i) \end{aligned} \tag{33}$$

where we recall quaternion $\boldsymbol{\rho}_i$ defines an arbitrary plane of encirclement. In classical examples of dynamic encirclement, all agents would encircle the target within a common plane [20], [22] and, in fact, the goal is to control the agents such that they remain in this plane. However, in some cases, it may be desirable to encircle a target in more that one plane of encirclement (*enspherement*) in order to improve sensor coverage or concentrate effects.

The formulation above is easily extended to support various overlapping planes of encirclement, forming a sparsely-populated sphere. This is accomplished at implementation, by defining planes of encirclement with different orientations $\boldsymbol{\rho}_i$. Let us define the set of all agents in such a sphere as $\mathcal{V}_\Omega$, composed of η_Ω planes of encirclement as follows:

$$\mathcal{V}_\Omega = \{\mathcal{V}_\theta^1, \mathcal{V}_\theta^2, \ldots, \mathcal{V}_\theta^{\eta_\Omega}\} \tag{34}$$

where $\mathcal{V}_\theta^1, \mathcal{V}_\theta^2, \ldots, \mathcal{V}_\theta^{\eta_\Omega}$ are planes of encirclement such that $\mathcal{V}_\theta^i \cap \mathcal{V}_\theta^j = \emptyset \ \forall \ i \neq j \in \{1, 2, \ldots, \eta_\Omega\}$. This previous statement enforces the physical constraint that no single agent can be assigned to multiple planes of encirclement. Therefore, we can define the indices for agents in the same plane of encirclement of agent i as:

$$N_i^{\theta_i} = \{m \in \mathcal{V}_\Omega : \boldsymbol{\rho}_i = \boldsymbol{\rho}_m\} \tag{35}$$

where we differentiate an enspherement (vice encirclement) formation as one in which not all agents in $\mathcal{V}_\Omega$ have the same values of $\boldsymbol{\rho}_i$. Note that these planes have overlapping segments around the target. Therefore, the obstacle avoidance term u_i^β in (18) is vitally important, as it serves to avoid collisions between agents at intersecting trajectories in these overlapping planes.

V. SMOOTH TRANSITION

This section presents a method for smoothly transitioning between the lattice flocking formation in Section III to the dynamic enspherement formation in Section. IV. We begin by defining a transition smoothing function $\tau(\cdot)$ bounded between 0 and 1 by a continuous sigmoid function. The function is dependent on the location of the agent q_i, the target q_r, and some desired transition distance from the target d_τ as follows:

$$\tau(z_\tau(t)) = \frac{1}{1+e^{-\omega_\tau z_\tau(t)}} \tag{36}$$

where

$$z_\tau(t) = (||q_i - q_r||_\sigma - d_\tau) \tag{37}$$

varies with time t based on the location of the agent relative to the target and ω_τ modulates the rate of transition. Note that we compute (37) using the σ-norm $||\cdot||_\sigma$ because, whereas the Euclidean-norm $||\cdot||$ is not differentiable at $q_i - q_r = 0$, the σ-norm is differentiable everywhere. This will help formally demonstrate the smoothness of the transition. The function is depicted in Fig. 1 for different values of ω_τ.

For convenience, let us drop the explicit notation for the time dependence of $z_\tau(t)$ and simply use z_τ. We combine (9) and (18) to form a single, bimodal control algorithm that autonomously and smoothly transitions between lattice flocking and dynamic enspherement as:

$$u_i = \tau_i(z_\tau)(u_i^\alpha + u_i^\gamma) + (1-\tau_i(z_\tau))(u_i^\delta + u_i^\zeta) + u_i^\beta \tag{38}$$

where $\tau_i(\cdot)$ is computed using (36) for the ith agent, u_i is the control signal, as described in (4), for the ith agent. Assuming the agents start at a distance far from the target (i.e. $||q_i - q_r|| > d_\tau$), we see that $\tau(z_\tau)$ starts with a value near 1. When $\tau(z_\tau)$ is near 1, the agents focus almost exclusively on forming a lattice. As $\tau(z_\tau)$ approaches 0, the agents focus more on enspherement. The desired transition distance from the target d_τ occurs at $\tau(z_\tau) = 0.5$, during which time the agents are midway through transitioning between formations.

It is desirable for the transition in (38), as governed by the varying of $\tau_i(\cdot)$ with time, to be continuous (or, *smooth*). Striving for smoothness in agent motion is what motivates the continuous sigmoid-based collective potential functions in previous work [15]. Practically, a smooth transition reduces necessary perturbations and energy consumption during the transition between the stable tactics of flocking and dynamic encirclement/enspherement. Here we formally demonstrate the smoothness of the proposed technique.

In Theorem 1 we demonstrate that the proposed bimodal swarming technique is smooth. We define function $\tau(z_\tau)$ to be *smooth* if it is differentiable everywhere (i.e. all n derivatives of the function $\frac{d^n}{dt^n}\tau(z_\tau)$ exist).

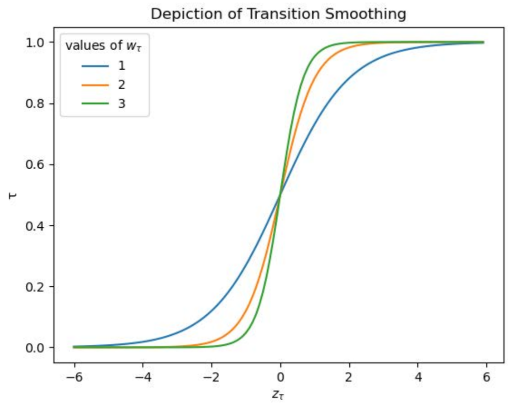


FIGURE 1. Depiction of transition smoothing function.

Theorem 1: Let us consider η agents with dynamics described by (4) *controlled by* (38). *If the transition between controller modes $\tau(z_\tau(t))$ is defined using* (36), *such that the value of d_τ is fixed and positive real, then $\tau(z_\tau(t))$ is differentiable everywhere and hence smooth.*

Proof: We begin by defining an expression for the nth derivative $\tau(z_\tau(t))$ with respect to time t using the generalized Faà di Bruno's formula as follows [31], [32]:

$$\frac{d^n}{dt^n}\tau(z_\tau) = \sum \frac{n!}{k_1!\cdots k_n!}\,\frac{d^m}{dz_\tau^m}\tau(z_\tau)\Big(\frac{d}{dt}\frac{z_\tau}{1!}\Big)^{k_1} \times \cdots \Big(\frac{d^n}{dt^n}\frac{z_\tau}{n!}\Big)^{k_n} \tag{39}$$

where the sum is taken over positive integers $\{k_1, \ldots, k_n\}$ such that $\sum_{i=1}^{n} k_i = m$; $k_1 + 2k_2 + \cdots + nk_n = n$; and we use compacted notation z_τ for $z_\tau(t)$. Intuitively, Faà di Bruno's formula is essentially a generalization of the common chain rule to n derivatives. As stated in the Assumptions for Theorem I of [32], a necessary condition for (39) to be true is that both $\tau(z_\tau)$ and z_τ are differentiable everywhere. Therefore, in order to show that (39) exists (and hence demonstrate that $\tau(z_\tau)$ is differentiable everywhere), we will show that all derivatives of composite functions $\tau(z_\tau)$ with respect to z_τ and z_τ with respect to t exist.

Let us consider the first composite function, the derivatives of $\tau(z_t)$ with respect to z_t and compute its first derivative:

$$\begin{aligned}
\frac{d\tau(z_\tau)}{dz_\tau} &= \frac{d}{dz_\tau}\Big(\frac{1}{1+e^{-z_\tau}}\Big) \\
&= \frac{d}{dz_\tau}(1+e^{-z_\tau})^{-1} \\
&= -(1+e^{-z_\tau})^{-2}\cdot(-e^{-z_\tau}) \\
&= \frac{e^{-z_\tau}}{(1+e^{-z_\tau})^2} \\
&= \frac{1}{1+e^{-z_\tau}}\cdot\frac{e^{-z_\tau}}{1+e^{-z_\tau}} \\
&= \frac{1}{1+e^{-z_\tau}}\cdot\frac{e^{-z_\tau}+1-1}{1+e^{-z_\tau}} \\
&= \frac{1}{1+e^{-z_\tau}}\cdot\Big(\frac{1+e^{-z_\tau}}{1+e^{-z_\tau}} - \frac{1}{1+e^{-z_\tau}}\Big)
\end{aligned}$$

$$= \frac{1}{1+e^{-z_\tau}} \cdot (1 - \frac{1}{1+e^{-z_\tau}})$$

which is more compactly expressed as

$$\frac{d\tau(z_\tau)}{dz_\tau} = \tau(z_\tau) \cdot (1 - \tau(z_\tau)) \tag{40}$$

The explicit derivation of (40) reveals the convenient properties of the exponential function that underpin the smoothness of (38); namely, that it is its own derivative ($\frac{de^x}{dx} = e^x$). Drawing from related work - specifically, Lemma 9 from the seminal work on sigmoid derivatives in [33] - we expand (40) for the general case of the nth derivative of $\tau(z_\tau)$ with respect to z_τ as follows:

$$\frac{d^n}{dz_\tau^n}\tau(z_\tau) = \sum_{k=1}^{n}(-1)^{k-1}C_k^{(n)}(\tau(z_\tau))^k(1-\tau(z_\tau))^{n+1-k} \tag{41}$$

where coefficients $C_k^{(n)}$ are for different values of n given in Table 1 of [33]. These results demonstrate that $\tau(z_\tau)$ is differentiable everywhere with respect to z_τ and partially satisfies the conditions for (39).

The remaining composite function is z_τ, which varies with t and represents the dynamics of the agent. Recall from (4) that these dynamics are, by definition, continuous. Also recall that, as initially proposed for multiagent flocking in [15], z_τ is computed using the σ-norm rather than the Euclidean norm in (37) to ensure it is differentiable everywhere. Therefore, if we assume q_i evolves according to the continuous (i.e. differentiable everywhere) double-integrator described in (4), then z_τ is also differentiable everywhere.

Finally, since both $\tau(z_\tau)$ and z_τ are differential everywhere, the nth derivate in (39) exists and the transition presented in (38) is smooth. □

VI. RESULTS

The bimodal dynamic swarming technique described above was implemented in two simulations. The agent dynamics were simulated using the open-source Python NumPy library, which provides a collection of common high-level mathematical functions for application on multi-dimensional arrays and matrices. The swarming parameters common to all simulations are summarized in Table 1.

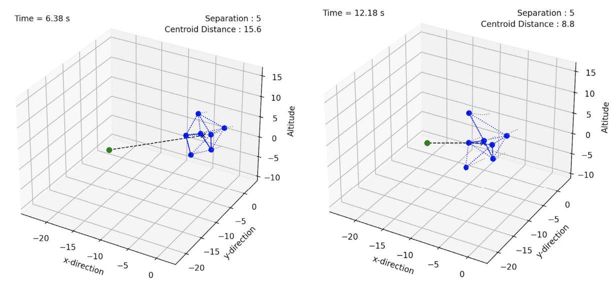
In the results for the first simulation, we demonstrate the positive benefits of blending the control inputs, as proposed in (38), when transitioning between lattice flocking and dynamic encirclement in a single plane. In the second simulation, we present results for a transition from lattice flocking to dynamic enspherement.

A. TRANSITION TO ENCIRCLEMENT

In this section, we present the results of blending the control inputs, as proposed in (38), when transitioning between lattice flocking and dynamic encirclement in the xy-plane. As shown in Fig. 2a, 7 agents were randomly distributed around the origin with an initial goal of forming a lattice with separation 5 m. The target was initiated at position $(-15, -15, 2)$ and translated towards the origin at a constant rate of 0.1 ms^{-1}. The swarm was programmed to track the target in a lattice formation and transition to encirclement at a radius of 5 m around the target with $d_\tau = 10$ m. In Fig. 2b we see that, as expected, the lattice formation gradually breaks away as $\tau(z_\tau)$ approaches 0.5 (i.e. the agents gets closer to d_τ).

TABLE 1. Parameters of the swarm.

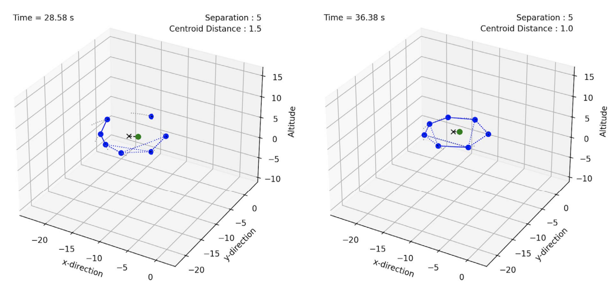
Parameter	Value
d	5
r	10
d'	1
r'	2
h	0.9
$c_1^\alpha, c_1^\beta, c_1^\delta$	2
$c_2^\alpha, c_2^\beta, c_2^\delta$	2.8
c_1^γ	1
c_2^γ	2
ω_τ	0.5



(a) (b)

FIGURE 2. Simulaton 1- Lattice stage: (a) Lattice, (b) Transition from lattice.

In Fig. 3a we see a gradual transition to encirclement as $\tau(z_\tau)$ approaches 0 (i.e. the agents each pass d_τ). In Fig. 3b, we see that the swarm has successfully adopted an encirclement formation around the target.



(a) (b)

FIGURE 3. Simulaton 1- Encirclement stage: (a) Transition to encirclement, (b) Encirclement.

The benefits of the smooth transition using $\tau(z_\tau)$ are illustrated in Fig. 4. Here, we compare the proposed *blended* tactic against an instantaneous *switched* tactic like the one used in [27]. Notice that at the center of the tactic change (approximately 12 s), the *switched* tactic produces a dramatic spike in commands, whereas the *blended* tactic produces only a minor change. As a result, the centroid of the entire swarm arrives over the target more quickly while consuming less energy in the transition.

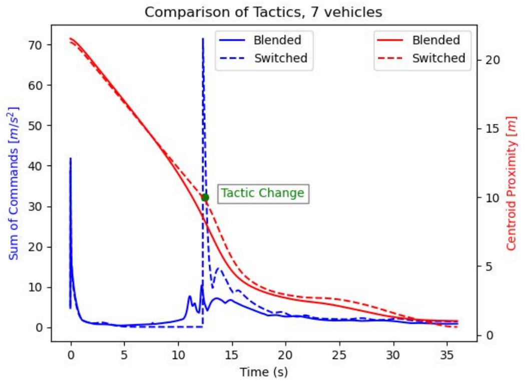


FIGURE 4. Comparison of blended versus switched tactics.

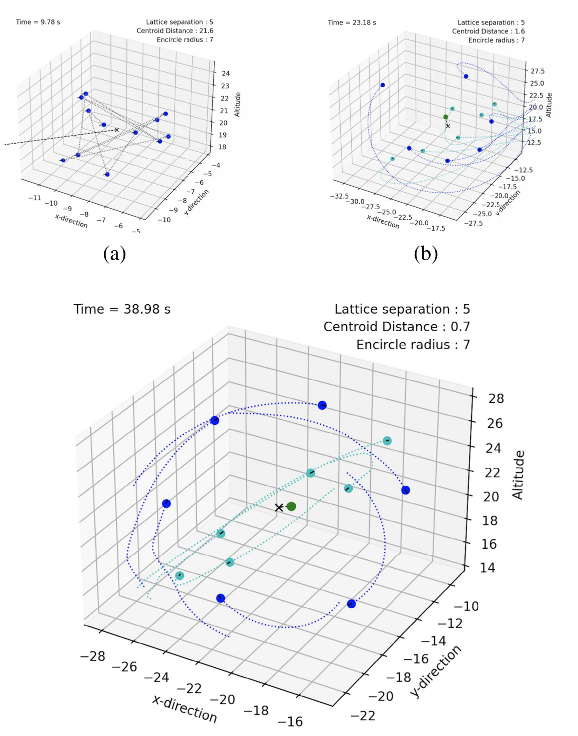


FIGURE 5. Simulation results for the network: (a) Lattice, (b) Transition between formations, (c) Enspherement.

B. ENSPHEREMENT

In this section, we present the results of blending the control inputs, as proposed in (38), when transitioning between lattice flocking and dynamic enspherement in arbitrary planes. Recall from Section IV-D that enspherement is accomplished by rotating the planes of encirclement for agent i by quaternion $\boldsymbol{\rho}_i$. Here, we defined two overlapping planes rotated by $\frac{\pi}{3}$ around the x-axis and $-\frac{\pi}{4}$ around the y-axis, respectively.

As shown in Fig. 5a, 12 agents were randomly distributed around the origin with an initial goal of forming a lattice flock with separation 5 m between agents. The target was initiated at position $(-25, -25, 20)$ and translated towards the origin at a constant rate of 0.1 ms^{-1}. The swarm was programmed to track the target in the lattice formation and transition to enspherement at a radius of 7 m with $d_\tau = 15$ m. In Fig. 5b we see the agents break from the lattice flocking formation to take up positions in a dynamic sphere.

In Fig. 5c, we see the swarm adopts dynamic enspherement around the target. At the intersections between the planes, we observe small perturbations from the ideal encirclement. This is due to the collision avoidance term.

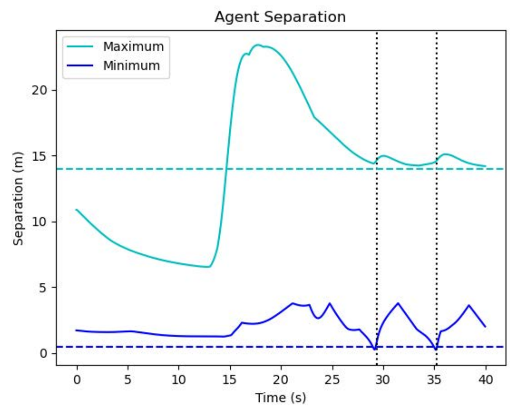


FIGURE 6. Illustration of minimum and maximum separation between vehicles throughout the experiment.

Fig. 6 presents a plot of the minimum and maximum separation between vehicles throughout the experiment. In the maximum separation (upper portion of the plot), we see that the agents settle at a separation of approximately 14 m (i.e. the diameter of the sphere). Perturbations are observed when collision avoidance was necessary. Vertical lines denote times when two agents approached each other at intersection planes and were forced to alter their trajectories. Note that this occurs at times of minimum separation (lower portion of the graph).

VII. CONCLUSION

This paper presented a bimodal control strategy for smoothly transitioning between different topologies of flocking and a new structured swarming technique called dynamic enspherement. We built on previous work by formulating dynamic encirclement for application in arbitrary planes using quaternions. We structured this formulation in terms of Reynolds rules so that they could be integrated with common graph-based flocking strategies. We formulated a new type of structured swarming that captures a target within a dynamic sphere by defining overlapping planes of encirclement. Finally, we implemented these new techniques in realistic simulations that demonstrated the effectiveness of enspherement and smoothness of the transitions between formations. When compared to tactic switching techniques found in previous work, the proposed approach brought the centroid of the entire swarm over the target more quickly while consuming less energy in the transition. This technique could prove useful in applications where agents must first pursue and close in on a target prior to carrying out a mission. Future work will focus on transitions between more sophisticated, time-varying structures.

REFERENCES

[1] J. A. Fax and R. M. Murray, ''Information flow and cooperative control of vehicle formations,'' *IEEE Trans. Autom. Control*, vol. 49, no. 9, pp. 1465–1476, Sep. 2004.

[2] X. Dong, Y. Zhou, Z. Ren, and Y. Zhong, ''Time-varying formation tracking for second-order multi-agent systems subjected to switching topologies with application to quadrotor formation flying,'' *IEEE Trans. Ind. Electron.*, vol. 64, no. 6, pp. 5014–5024, Jun. 2017.

[3] J. Toner and Y. Tu, ''Flocks, herds, and schools: A quantitative theory of flocking,'' *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 58, no. 4, pp. 4828–4858, Oct. 1998. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevE.58.4828

[4] N. Shimoyama, K. Sugawara, T. Mizuguchi, Y. Hayakawa, and M. Sano, ''Collective motion in a system of motile elements,'' *Phys. Rev. Lett.*, vol. 76, no. 20, pp. 3870–3873, May 1996, doi: 10.1103/PhysRevLett.76.3870.

[5] A. Mogilner and L. Edelstein-Keshet, ''A non-local model for a swarm,'' *J. Math. Biol.*, vol. 38, no. 6, pp. 534–570, Jun. 1999.

[6] C. W. Reynolds, ''Flocks, herds and schools: A distributed behavioral model,'' in *Proc. 14th Annu. Conf. Comput. Graph. Interact. Techn. (SIGGRAPH)*. New York, NY, USA: Association for Computing Machinery, 1987, pp. 25–34, doi: 10.1145/37401.37406.

[7] K. Saulnier, D. Saldaña, A. Prorok, G. J. Pappas, and V. Kumar, ''Resilient flocking for mobile robot teams,'' *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 1039–1046, Apr. 2017.

[8] K. D. Do, ''Flocking for multiple elliptical agents with limited communication ranges,'' *IEEE Trans. Robot.*, vol. 27, no. 5, pp. 931–942, Oct. 2011.

[9] L. A. V. Reyes and H. G. Tanner, ''Flocking, formation control, and path following for a group of mobile robots,'' *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 4, pp. 1268–1282, Jul. 2015.

[10] W. Li and M. W. Spong, ''Analysis of flocking of cooperative multiple inertial agents via a geometric decomposition technique,'' *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 44, no. 12, pp. 1611–1623, Dec. 2014.

[11] N. Moshtagh and A. Jadbabaie, ''Distributed geodesic control laws for flocking of nonholonomic agents,'' *IEEE Trans. Autom. Control*, vol. 52, no. 4, pp. 681–686, Apr. 2007.

[12] T. Ibuki, S. Wilson, J. Yamauchi, M. Fujita, and M. Egerstedt, ''Optimization-based distributed flocking control for multiple rigid bodies,'' *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 1891–1898, Apr. 2020.

[13] R. Olfati-Saber and R. M. Murray, ''Consensus problems in networks of agents with switching topology and time-delays,'' *IEEE Trans. Autom. Control*, vol. 49, no. 9, pp. 1520–1533, Sep. 2004.

[14] R. Olfati-Saber, J. A. Fax, and R. M. Murray, ''Consensus and cooperation in networked multi-agent systems,'' *Proc. IEEE*, vol. 95, no. 1, pp. 215–233, Jan. 2007.

[15] R. Olfati-Saber, ''Flocking for multi-agent dynamic systems: Algorithms and theory,'' *IEEE Trans. Autom. Control*, vol. 51, no. 3, pp. 401–420, Mar. 2006.

[16] L. Zhang, Z. Zhang, Y. Qiao, and X. Wei, ''Surrounding control in cooperative second-order agent networks,'' in *Proc. 43rd Annu. Conf. IEEE Ind. Electron. Soc. (IECON)*, Oct. 2017, pp. 5604–5609.

[17] T.-H. Kim and S. Hara, ''Cyclic pursuit strategy for multi-agent dynamical systems with sampled communication,'' in *Proc. SICE Annu. Conf.*, Aug. 2008, pp. 2436–2441.

[18] T.-H. Kim and T. Sugie, ''Cooperative control for target-capturing task based on a cyclic pursuit strategy,'' *Automatica*, vol. 43, no. 8, pp. 1426–1431, 2007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S000510980700132X

[19] A. S. Matveev, A. A. Semakova, and A. V. Savkin, ''Tight circumnavigation of multiple moving targets based on a new method of tracking environmental boundaries,'' *Automatica*, vol. 79, pp. 52–60, May 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0005109817300511

[20] F. Dong, K. You, and S. Song, ''Target encirclement with any smooth pattern using range-based measurements,'' *Automatica*, vol. 116, Jun. 2020, Art. no. 108932. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0005109820301308

[21] J. Ma, H. Lu, J. Xiao, Z. Zeng, and Z. Zheng, ''Multi-robot target encirclement control with collision avoidance via deep reinforcement learning,'' *J. Intell. Robot. Syst.*, vol. 99, no. 2, pp. 371–386, Aug. 2020, doi: 10.1007/s10846-019-01106-x.

[22] A. T. Hafez, A. J. Marasco, S. N. Givigi, M. Iskandarani, S. Yousefi, and C. A. Rabbath, ''Solving multi-UAV dynamic encirclement via model predictive control,'' *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 6, pp. 2251–2265, Nov. 2015.

[23] R. Olfati-Saber, ''Swarms on sphere: A programmable swarm with synchronous behaviors like oscillator networks,'' in *Proc. 45th IEEE Conf. Decis. Control*, Dec. 2006, pp. 5060–5066.

[24] J. Hu, Y. Hong, and L. Gao, ''Tracking control for multi-agent consensus with an active leader and variable topology,'' *Automatica*, vol. 42, no. 7, pp. 1177–1182, Jul. 2007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0005109806001063

[25] D. Yu and C. L. P. Chen, ''Smooth transition in communication for swarm control with formation change,'' *IEEE Trans. Ind. Informat.*, vol. 16, no. 11, pp. 6962–6971, Nov. 2020.

[26] G. Wen, Y. Yu, Z. Peng, and A. Rahmani, ''Consensus tracking for second-order nonlinear multi-agent systems with switching topologies and a time-varying reference state,'' *Int. J. Control*, vol. 89, no. 10, pp. 2096–2106, 2016, doi: 10.1080/00207179.2016.1149221.

[27] A. T. Hafez, S. N. Givigi, H. M. Schwartz, S. Yousefi, and M. Iskandarani, ''Real time tactic switching for multiple cooperative UAVs via model predictive control,'' in *Proc. Annu. IEEE Syst. Conf. (SysCon)*, Apr. 2015, pp. 432–438.

[28] L. Dong and S. K. Nguang, ''Multiagent systems with continuously switching topologies,'' in *Consensus Tracking of Multi-Agent Systems With Switching Topologies* (Emerging Methodologies and Applications in Modelling), L. Dong and S. K. Nguang, Eds. New York, NY, USA: Academic, 2020, ch. 2, pp. 29–46. [Online]. Available: http://www.sciencedirect.com/science/article/pii/B9780128183656000021

[29] J. M. Chappell, A. Iqbal, J. G. Hartnett, and D. Abbott, ''The vector algebra war: A historical perspective,'' *IEEE Access*, vol. 4, pp. 1997–2004, 2016.

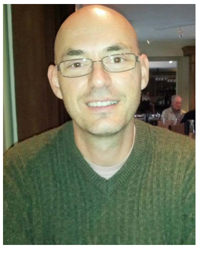
[30] O. Khatib, ''The potential field approach and operational space formulation in robot control,'' in *Adaptive and Learning Systems*. Boston, MA, USA: Springer, 1986, pp. 367–377.

[31] L. H. Encinas and J. M. Masqué, "A short proof of the generalized faà di Bruno's formula," *Appl. Math. Lett.*, vol. 16, no. 6, pp. 975–979, Aug. 2003. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0893965903900267

[32] L. E. Fraenkel, ''Formulae for high derivatives of composite functions,'' *Math. Proc. Cambridge Phil. Soc.*, vol. 83, no. 2, pp. 159–165, Mar. 1978.

[33] A. A. Minai and R. D. Williams, ''On the derivatives of the sigmoid,'' *Neural Netw.*, vol. 6, no. 6, pp. 845–853, 1993. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0893608005801297

PETER TRAVIS JARDINE (Member, IEEE) received the Ph.D. degree in electrical and computer engineering from Queen's University, Kingston, ON, Canada, in 2018. He is currently an Adjunct Assistant Professor with the Royal Military College of Canada. His research interests include the application of artificial intelligence in robotics and control.



SIDNEY N. GIVIGI, JR. (Senior Member, IEEE) received the Ph.D. degree in electrical and computer engineering from Carleton University, Ottawa, ON, Canada. He is currently an Associate Professor with the School of Computing, Queen's University, Kingston, ON, Canada. His research interests include machine learning, autonomous systems, and robotics.