2023 American Control Conference (ACC)
San Diego, CA, USA. May 31 - June 2, 2023

Constraint-Driven Optimal Control for Emergent Swarming and Predator Avoidance

Logan E. Beaver, *Member, IEEE*, Andreas A. Malikopoulos, *Senior Member, IEEE*

Abstract— **In this article, we present a constraint-driven optimal control framework that achieves emergent cluster flocking within a constrained 2D environment. We formulate a decentralized optimal control problem that includes safety, flocking, and predator avoidance constraints. We explicitly derive conditions for constraint compatibility and propose an event-driven constraint relaxation scheme. We map this to an equivalent switching system that intuitively describes the behavior of each agent in the system. Instead of minimizing control effort, as it is common in the ecologically-inspired robotics literature, in our approach, we minimize each agent's deviation from their most efficient locomotion speed. Finally, we demonstrate our approach in simulation both with and without the presence of a predator.**

I. Introduction

Multi-agent systems have attracted considerable attention in many applications due to their natural parallelization, general adaptability, and ability to self-organize [1]. One emerging application of multi-agent systems is mimicking the aggregate motion of certain birds and fish, also known as cluster flocking or swarming [2]. There are several purported advantages of cluster flocking in biological systems, including predator avoidance and estimating population size [3].

In this article, we derive a distributed control algorithm that induces cluster flocking in a multi-agent system. Prior work has primarily relied on reinforcement learning to achieve predator avoidance, including a multi-level approaches [4] and policy sharing [5], [6]. Traditional control approaches tend to achieve swarming behavior by implementing Reynolds flocking rules using potential fields [2]. These approaches have two major drawbacks. First, they inevitably drive agents into a regular lattice formation [7], which is not conductive to swarming. Second, potential fields are known to cause steady oscillation in agent trajectories and exacerbate deadlock in constrained environments [8].

In contrast to existing approaches, we propose a biologically-inspired approach based on an analysis of sand-eel schools in the presence of predators [9]. In this article, we build on our previous work with set-theoretic control [10]–[12], where we embed inter-agent and environmental interactions as state and control constraints in an optimal control problem. Our set-theoretic approach has the advantage of being interpretable, i.e., the cause of an agents' action can be deduced by examining which constraints are currently active. Our technical results are closely related to the control barrier function (CBF) literature, particularly multi-agent CBFs [13]. However, our approach does not require the constraints to be encoded as sub-level sets of a continuous function—we work with the sets directly. We also propose a solution to the open problem of constraint incompatibility through an event-triggered constraint relaxation scheme. Finally, we present a mapping between constraint-driven control and switching systems, which provides a rigorous and interpretable description of each boids' behavior. The contributions of this article are: 1) a decentralized optimal control algorithm that yields emergent swarming behavior, 2) an event-triggered scheme to selectively relax constraints and guarantee feasibility, 3) a rigorous mapping between constraint-driven control and switching systems, and 4) simulation results demonstrating emergent cluster flocking and predator avoidance behaviors.

The remainder of the article is organized as follows. In Section II, we formulate the cluster flocking problem and discuss our working assumptions. In Section III, we derive our optimal control policy, derive the safe action sets, and map the problem to a switching system. In Section IV, we validate our results in two simulations with 15 boids; the first demonstrates emergent cluster flocking, and the second demonstrates predator avoidance. Finally, we draw conclusions and propose some directions for future research in Section V.

II. Problem Formulation

We consider a set of $N \in \mathbb{N}$ boids indexed by the set $\mathcal{B} = \{1, 2, \dots, N\}$. Each boid $i \in \mathcal{B}$ obeys second-order integrator dynamics,

$$\begin{aligned}\dot{\boldsymbol{p}}_i(t) &= \boldsymbol{v}_i(t),\\ \dot{\boldsymbol{v}}_i(t) &= \boldsymbol{u}_i(t),\end{aligned} \tag{1}$$

where $\boldsymbol{p}_i(t), \boldsymbol{v}_i(t) \in \mathbb{R}^2$ correspond to the position and velocity of each boid, and $\boldsymbol{u}_i(t) \in \mathbb{R}^2$ is the control input. We also impose the state and control constraints,

$$\boldsymbol{p}_i(t) \in \mathcal{P}, \tag{2}$$

$$\boldsymbol{u}_i(t) \in \mathcal{U}, \tag{3}$$

where $\mathcal{P} \subset \mathbb{R}^2$ is a non-empty intersection of half-planes and $\mathcal{U} = \{u(t) \;:\; ||u||_\infty \leq u_{\max}\}$ ensures the boids' do not exceed their maximum control input at any time instant. We employ the infinity norm to simplify our mathematical exposition; however, the norm does not impose any restrictions in our approach.

This research was supported by NSF under Grants CNS-2149520 and CMMI-2219761.

L.E. Beaver is with the Division of Systems Engineering, Boston University, Boston, MA, USA 02215 (email: lebeaver@bu.edu).

A.A. Malikopoulos is with the Department of Mechanical Engineering, University of Delaware, Newark, DE, USA 19716 (email: andreas@udel.edu).

We account for interactions between boids using Voronoi tessellation [14]. Under this approach, each boid is considered the center of a Voronoi cell. We define the interaction graph $\mathcal{V}(t) \subset \mathcal{B} \times \mathcal{B}$ which contains edges (i, j) and (j, i) at time t if and only if the Voronoi cells i and j share a common edge at that time. Equivalently, the set $\mathcal{V}(t)$ is the Delaunay triangulation of the boids' positions.

Definition 1 (Voronoi Neighborhood)**.** The neighborhood of each boid $i \in \mathcal{B}$ is the set,

$$\mathcal{N}_i(t) \coloneqq \big\{ j \in \mathcal{B} \ : \ (i, j) \in \mathcal{V}(t) \big\}, \tag{4}$$

where boid i can receive information, via communication or sensing, with any other boid $j \in \mathcal{N}_i(t)$.

As with $k-$nearest neighbors, a Voronoi neighborhood may allow the boids to sense neighbors that are arbitrarily far away in general. Similar to past work [10], [15], we do not presume the boids possess infinite sensing capabilities; rather that Definition 1 describes the interactions between boids over their relatively small separating distances. One potential solution is to only consider Voronoi neighbors that are within a fixed sensing range [14], although results from biology demonstrate that this is, in general, unnecessary [16].

Our objective is to generate emergent swarming behavior, such that the boids remain close to their neighbors to avoid predators. To achieve an aggregate swarming motion, we implement a variation of the disk flocking constraint proposed in [10]. First, we determine the neighborhood center for each boid $i \in \mathcal{B}$,

$$\boldsymbol{c}_i(t) = \frac{1}{|\mathcal{N}_i(t)|} \sum_{j \in \mathcal{N}_i(t)} \boldsymbol{p}_j(t). \tag{5}$$

Note that Definition 1 guarantees $|\mathcal{N}_i(t)| > 0$. We use the neighborhood center to construct the relative position vector,

$$\boldsymbol{r}_i(t) \coloneqq \boldsymbol{p}_i(t) - \boldsymbol{c}_i(t). \tag{6}$$

Finally, to achieve swarming, we require each boid i to approach and remain within a distance $R \in \mathbb{R}_{>0}$ of the neighborhood center, i.e.,

$$g_i(\boldsymbol{r}_i(t), \boldsymbol{u}_i(t)) = \begin{cases} ||\boldsymbol{r}_i(t)|| - R & \text{if } ||\boldsymbol{r}_i(t)|| \leq R, \\ \frac{||\dot{\boldsymbol{r}}_i(t)||}{u_{\max}} \boldsymbol{u}_i(t) \cdot \boldsymbol{r}_i(t) + \dot{\boldsymbol{r}}_i(t) \cdot \boldsymbol{r}_i(t) & \text{o.w.}, \end{cases}$$
$$g_i(\boldsymbol{r}_i(t), \boldsymbol{u}_i(t)) \leq 0,$$

where '$\cdot$' denotes the dot product, i.e., $\boldsymbol{u} \cdot \boldsymbol{r}_i = \boldsymbol{u}^\intercal \boldsymbol{r}_i$. Note that the first case is trivially satisfied, i.e., the boid must remain within the disk while inside the disk. Thus, we write

$$||\boldsymbol{r}_i(t)|| > R \implies \frac{||\dot{\boldsymbol{r}}_i(t)||}{u_{\max}} \boldsymbol{u}_i(t) \cdot \boldsymbol{r}_i(t) + \dot{\boldsymbol{r}}_i(t) \cdot \boldsymbol{r}_i(t) \leq 0. \tag{7}$$

We emphasize that our objective is not to trap boid i within the disk of radius R centered at $\boldsymbol{c}_i(t)$. Instead, we expect the switching neighborhood topology and dynamic motion of $\boldsymbol{c}_i(t)$ to drive the swarming behavior. Additionally, the form of (7) is inspired by energy-saving techniques in [17]. Note that when boid i travels in the "correct" direction, i.e., $\dot{\boldsymbol{r}}_i(t) \cdot \boldsymbol{r} < 0$, the control action $\boldsymbol{u}_i(t)$ can take some values in the same direction as $\boldsymbol{r}_i(t)$. However, when boid i is traveling in the "wrong" direction, i.e., $\dot{\boldsymbol{r}}_i(t) \cdot \boldsymbol{r}_i(t) > 0$, the control action $\boldsymbol{u}_i(t)$ must be at least partially opposed to $\boldsymbol{r}_i(t)$ to drive boid i toward $\boldsymbol{c}_i(t)$.

Next, inspired by the empirical data collected on sand-eels [9], we model the predator as a ball of radius Γ. We define the relative distance vector between each boid i and the predator as,

$$\boldsymbol{d}_i(t) \coloneqq \boldsymbol{p}_i(t) - \boldsymbol{o}(t), \tag{8}$$

where $\boldsymbol{o}(t)$ is the position of the predator at time t. To ensure predator avoidance, we select a value of Γ larger than the diameter of the predator and employ a similar constraint to repel the boids,

$$||\boldsymbol{d}_i(t)|| < \Gamma \implies - \frac{||\dot{\boldsymbol{d}}_i(t)||}{u_{\max}} \boldsymbol{u}_i(t) \cdot \boldsymbol{d}_i(t) - \dot{\boldsymbol{d}}_i(t) \cdot \boldsymbol{d}_i(t) \leq 0. \tag{9}$$

With the constraints defined, our next objective is to design an optimal control problem such that the individual boid motion generates emergent swarming behavior. To this end, we impose the following assumptions on our system.

Assumption 1. Each boid is equipped with a low-level controller that is capable of tracking the control input.

We impose Assumption 1 to simplify our analysis and understand how the system performs in the ideal case. Assumption 1 is common for trajectory generation problems, and it can be relaxed by introducing robust control terms or a safety layer, e.g., using a control barrier function [18].

Assumption 2. The boids have sufficient vertical space to avoid collisions between each other without an explicit collision-avoidance constraint.

Assumption 2 is common in 2D swarming applications [6], [19]. Furthermore, it has been thoroughly demonstrated that adding an extra dimension of motion can significantly reduce the likelihood of collisions [20].

III. Solution Approach

We employ constraint-driven control to generate the control input for each boid. This is an optimization approach wherein the desired behavior of each boid is encoded as a constraint in an optimal control problem. This technique has been used successfully to control multi-agent systems [13], [21], [22]. Each boid solves the optimal control problem reactively, i.e., they take an action at each time-instant and do not account for the system state at future time steps. Our motivation for this is twofold: first, it overcomes the computational and communication costs associated with decentralized trajectory planning [2]. Second, it allows boids to freely enter and leave the domain, e.g., due to operating constraints, mechanical failure, or predation, as the constraint boundaries are a function of the local system state. For the remainder of our exposition, we omit the explicit dependence of state variables on t when no ambiguity arises.

We start with the position constraint (2), which is not an explicit function of the control input. Let $k = 1, 2, \dots, M$ index the M hyperplanes that define the boundary of $\mathcal{P}$. Each hyperplane $k = 1, 2, \dots, M$ consists of a normal vector $\hat{\boldsymbol{n}}_k \in \mathbb{R}^2$ and offset $b_k \in \mathbb{R}$; the signed distance to the surface of hyperplane k is,

$$d_{ik} = \boldsymbol{p}_i \cdot \hat{\boldsymbol{n}}_k + b_k, \tag{10}$$

for boid $i \in \mathcal{B}$. Note that our convention assumes the normal vector $\boldsymbol{n}_k$ points away from the feasible region $\mathcal{P}$. To guarantee constraint satisfaction, we require the time derivative of (10) to be non-positive when the constraint is active, i.e.,

$$\boldsymbol{p}_i \cdot \hat{\boldsymbol{n}}_k + b_k = 0 \implies \boldsymbol{v}_i \cdot \hat{\boldsymbol{n}}_k \leq 0. \tag{11}$$

This safety constraint (11) can be achieved by using a stopping distance constraint for each $k = 1, 2, \dots, M$ [22],

$$g_{ik} = \Big(\boldsymbol{p}_i \cdot \hat{\boldsymbol{n}}_k + b_k\Big) + \alpha \frac{\Big(\boldsymbol{v}_i \cdot \hat{\boldsymbol{n}}_k\Big)^2}{2u_{\max}} \leq 0, \tag{12}$$

where $\alpha \in \mathbb{R}_{>0}$ is a parameter that determines the stopping distance. Note that (12) trivially satisfies (11). This leads to our definition of the safe action set.

Definition 2 (Safe Action Set)**.** For each boid $i \in \mathcal{B}$ at time t, the safe action set is,

$$\begin{aligned}\mathcal{A}_i^s \coloneqq \Big\{ & \boldsymbol{u}_i \in \mathbb{R}^2 \;:\; ||\boldsymbol{u}_i||_\infty - u_{\max} \leq 0, \\ & \Big(\boldsymbol{p}_i \cdot \hat{\boldsymbol{n}}_k + b_k\Big) + \alpha \frac{\Big(\boldsymbol{v}_i \cdot \hat{\boldsymbol{n}}_k\Big)^2}{2u_{\max}} = 0 \implies \\ & \boldsymbol{v}_i \cdot \hat{\boldsymbol{n}}_k \Big(1 + \frac{\alpha}{u_{\max}}(\boldsymbol{u}_i \cdot \hat{\boldsymbol{n}}_k)\Big) \leq 0, \\ & \forall k = 1, 2, \dots, M \Big\}.\end{aligned}$$

In our approach, we constrain the boids to remain within an axis-aligned rectangular domain, i.e., $\mathcal{P}$ is constructed from two pairs of parallel hyperplanes that intersect at right angles. With the safe action set and constraints defined, each boid i also requires a notion of performance to select the "best" control input. The ecologically-inspired paradigm [23] would suggest minimizing the norm of the control input, which arguably yields a minimum effort policy. However, as discussed in [9], sand-eels tend to cruise at a constant speed of approximately 2 body lengths per second. Thus, we require each boid to match an optimal swimming speed, denoted $||\boldsymbol{v}_i^*||$, as closely as possible, i.e.,

$$J_i\big(\boldsymbol{v}_i(t)\big) = \Big(||\boldsymbol{v}_i(t)|| - ||\boldsymbol{v}_i{}^*||\Big)^2. \tag{13}$$

We interpret the optimal swimming speed as being biomechanically advantageous, i.e., if $J = ||\boldsymbol{u}_i||$ minimizes energy consumption, then (13) corresponds to minimum-power locomotion.

Finally, while we analyze the system in continuous-time, the optimal control problem must be implemented in discrete time. This is a practical consideration for implementation on a digital computer, and it is necessary here because the objective is a function of the states rather than the control input. Discretizing the optimal control problem is an area of open research, where potential solutions include letting the time step Δt grow to an infinitesimally small dt [24] or tightening the constraints at each time step to ensure the continuous trajectory is feasible [25].

Problem 1. For each boid $i \in \mathcal{B}$ at time t, apply the control action that solves,

$$\min_{\boldsymbol{u}_i(t)} \Big(||\boldsymbol{v}_i(t) + \boldsymbol{u}_i(t)\,\Delta t|| - ||\boldsymbol{v}_i{}^*||\Big)^2$$

subject to:

$$\boldsymbol{u}_i(t) \in \mathcal{A}_i^s, \text{ (1), (7), (9).}$$

Next, we present a result that guarantees recursive feasibility for the safe action set. While this result relies on having an axis-aligned rectangular domain, an extension is straightforward, e.g., by replacing (25) with the L^2 norm of the control effort.

Theorem 1. *For a fixed value of $\alpha \geq 1$, if a boid $i \in \mathcal{B}$ satisfies (12) at some time t for a rectangular domain $\mathcal{P}$, then $\mathcal{A}_i^s$ satisfies recursive feasibility for all future time.*

Proof. Let the rectangular domain $\mathcal{P}$ consists of four hyperplanes, indexed by $k = 1, 2, 3, 4$ such that $\hat{\boldsymbol{n}}_1 = -\hat{\boldsymbol{n}}_3$ and $\hat{\boldsymbol{n}}_2 = -\hat{\boldsymbol{n}}_4$. Without loss of generality, let $\boldsymbol{v}_i \cdot \hat{\boldsymbol{n}}_1 > 0$ and $\boldsymbol{v}_i \cdot \hat{\boldsymbol{n}}_2 > 0$. When the safety constraint (12) is not active, boid i may take any action satisfying the control bounds (25). However, when (12) is active, we must ensure its derivative is non-positive to guarantee safety. Taking the derivative of (12) and combining terms yields,

$$\dot{g}_{ik} = \boldsymbol{v}_i \cdot \hat{\boldsymbol{n}}_k \Big(1 + \frac{\alpha}{u_{\max}}\big(\boldsymbol{u}_i \cdot \hat{\boldsymbol{n}}_k\big)\Big). \tag{14}$$

We seek a control input such that $\dot{g}_{ik} \leq 0$. For $k = 1, 2$, dividing by $\boldsymbol{v}_i \cdot \hat{\boldsymbol{n}}_k > 0$ yields a condition on $\boldsymbol{u}_i$,

$$\boldsymbol{u}_i \cdot \hat{\boldsymbol{n}}_1 \leq -\frac{u_{\max}}{\alpha}, \quad \boldsymbol{u}_i \cdot \hat{\boldsymbol{n}}_2 \leq -\frac{u_{\max}}{\alpha}. \tag{15}$$

Similarly, for $k = 3, 4$, dividing by $\boldsymbol{v}_i \cdot \hat{\boldsymbol{n}}_k < 0$ implies,

$$\boldsymbol{u}_i \cdot \hat{\boldsymbol{n}}_3 \geq -\frac{u_{\max}}{\alpha}, \quad \boldsymbol{u}_i \cdot \hat{\boldsymbol{n}}_4 \geq -\frac{u_{\max}}{\alpha}. \tag{16}$$

Substituting $\hat{\boldsymbol{n}}_1 = -\hat{\boldsymbol{n}}_3$ and $\hat{\boldsymbol{n}}_2 = -\hat{\boldsymbol{n}}_4$ into (16) yields the conditions,

$$\boldsymbol{u}_i \cdot \hat{\boldsymbol{n}}_1 \leq \frac{u_{\max}}{\alpha}, \quad \boldsymbol{u}_i \cdot \hat{\boldsymbol{n}}_2 \leq \frac{u_{\max}}{\alpha}. \tag{17}$$

Thus, to guarantee g_{ik} is nonincreasing, the control input must satisfy (15) and (17), i.e.,

$$\boldsymbol{u}_i \cdot \hat{\boldsymbol{n}}_1 \leq -\frac{u_{\max}}{\alpha} \leq \frac{u_{\max}}{\alpha}, \tag{18}$$

$$\boldsymbol{u}_i \cdot \hat{\boldsymbol{n}}_2 \leq -\frac{u_{\max}}{\alpha} \leq \frac{u_{\max}}{\alpha}. \tag{19}$$

This is satisfied by the candidate control action,

$$\boldsymbol{u}_i = -\frac{u_{\max}}{\alpha}\hat{\boldsymbol{n}}_1 - \frac{u_{\max}}{\alpha}\hat{\boldsymbol{n}}_2, \tag{20}$$

as $\hat{\boldsymbol{n}}_1 \cdot \hat{\boldsymbol{n}}_2 = 0$ by definition. In our axis-aligned domain, the control constraint implies,

$$||\boldsymbol{u}_i||_\infty = \max\left\{\frac{u_{\max}}{\alpha}, \frac{u_{\max}}{\alpha}\right\} = \frac{1}{\alpha}u_{\max}, \tag{21}$$

which satisfies (3) for $\alpha \geq 1$. Finally, for the case that $\boldsymbol{v}_i \cdot \hat{\boldsymbol{n}}_k = 0$ for any $k = 1, 2, 3, 4$, the corresponding derivative $\dot{\boldsymbol{g}}_{ik} = 0$ for every control input. □

Thus, given a feasible initial state, Theorem 1 guarantees that each boid's trajectory will remain feasible indefinitely if its control action is selected from $\mathcal{A}_i^s$. However, we require each boid to jointly satisfy the safety, swarming (7), and predator avoidance (9) constraints to achieve emergent cluster flocking behavior. Thus, guaranteeing the recursive feasibility of $\mathcal{A}_i^s$ is insufficient to ensure a feasible control action exists. The following results provide the explicit conditions for constraint incompatibility, i.e., when the set of feasible control actions becomes empty.

Lemma 1. For a boid $i \in \mathcal{B}$, let $k = 1, 2$ index two perpendicular hyperplanes in the rectangular domain such that $\boldsymbol{v}_i \cdot \hat{\boldsymbol{n}}_k \geq 0$. Then, if (12) is strictly equal to zero and $||\boldsymbol{r}_i|| > R_i$, there is no feasible action if none of the conditions,

$$||\dot{\boldsymbol{r}}_i||\left(\begin{bmatrix}1\\ \alpha^{-1}\\ 1\\ \alpha^{-1}\end{bmatrix}(\hat{\boldsymbol{n}}_1 \cdot \hat{\boldsymbol{r}}_i) + \begin{bmatrix}1\\ 1\\ \alpha^{-1}\\ \alpha^{-1}\end{bmatrix}(\hat{\boldsymbol{n}}_2 \cdot \hat{\boldsymbol{r}}_i)\right) \geq \dot{\boldsymbol{r}}_i \cdot \hat{\boldsymbol{r}}_i, \tag{22}$$

hold at time t for $k = 1, 2$.

Proof. Under the premise of Lemma 1, we must determine when the constraint,

$$\frac{||\dot{\boldsymbol{r}}_i||}{u_{\max}}\boldsymbol{u}_i \cdot \hat{\boldsymbol{r}}_i + \dot{\boldsymbol{r}}_i \cdot \hat{\boldsymbol{r}}_i \leq 0, \tag{23}$$

is incompatible with $\mathcal{A}_i^s$. First, $\dot{\boldsymbol{r}}_i = 0$ satisfies (23) for any $\boldsymbol{u}_i$, thus, we may divide (23) by $||\dot{\boldsymbol{r}}_i||$ and work with unit vectors for the remainder of the proof, i.e.,

$$\frac{1}{u_{\max}}\boldsymbol{u}_i \cdot \hat{\boldsymbol{r}}_i + \hat{\dot{\boldsymbol{r}}}_i \cdot \hat{\boldsymbol{r}}_i \leq 0. \tag{24}$$

Next, we consider the control $\boldsymbol{u}_i = -u_1\hat{\boldsymbol{n}}_1 - u_2\hat{\boldsymbol{n}}_2$. From the proof of Theorem 1, (3) and (15) imply that u_1 and u_2 must satisfy,

$$1 \geq \frac{u_1}{u_{\max}} \geq \frac{1}{\alpha}, \quad 1 \geq \frac{u_2}{u_{\max}} \geq \frac{1}{\alpha}. \tag{25}$$

The swarming constraint (24) becomes,

$$\frac{u_1}{u_{\max}}(\hat{\boldsymbol{n}}_1 \cdot \hat{\boldsymbol{r}}_i) + \frac{u_2}{u_{\max}}(\hat{\boldsymbol{n}}_2 \cdot \hat{\boldsymbol{r}}_i) \geq \hat{\dot{\boldsymbol{r}}}_i \cdot \hat{\boldsymbol{r}}_i. \tag{26}$$

The result follows from substituting the bounds (25) into (26). □

Lemma 2. For a boid $i \in \mathcal{B}$, let $k = 1, 2$ index two perpendicular hyperplanes in the rectangular domain such that $\boldsymbol{v}_i \cdot \hat{\boldsymbol{n}}_k \geq 0$. Then, if (12) is strictly equal to zero and $||\boldsymbol{d}_i||<\Gamma$, there is no feasible action if none of the conditions,

$$||\dot{\boldsymbol{d}}_i||\left(\begin{bmatrix}1\\ \alpha^{-1}\\ 1\\ \alpha^{-1}\end{bmatrix}(\hat{\boldsymbol{n}}_1 \cdot \hat{\boldsymbol{d}}_i) + \begin{bmatrix}1\\ 1\\ \alpha^{-1}\\ \alpha^{-1}\end{bmatrix}(\hat{\boldsymbol{n}}_2 \cdot \hat{\boldsymbol{d}}_i)\right) \leq \dot{\boldsymbol{d}}_i \cdot \hat{\boldsymbol{d}}_i, \tag{27}$$

hold at time t for $k = 1, 2$.

Proof. The proof Lemma 2 is identical to Lemma 1, and thus we omit it. □

Lemma 3. For a boid $i \in \mathcal{B}$, let $k = 1, 2$ index two perpendicular hyperplanes in the rectangular domain such that $\boldsymbol{v}_i \cdot \hat{\boldsymbol{n}}_k \geq 0$. Then, if (12) is strictly equal to zero, $||\boldsymbol{r}_i|| > R_i$, and $||\boldsymbol{d}_i||<\Gamma$, there is no feasible control action if the linear inequalities,

$$\begin{bmatrix} ||\dot{\boldsymbol{r}}_i||\hat{\boldsymbol{n}}_1 \cdot \hat{\boldsymbol{r}}_i & ||\dot{\boldsymbol{r}}_i||\hat{\boldsymbol{n}}_2 \cdot \hat{\boldsymbol{r}}_i \\ -||\dot{\boldsymbol{d}}_i||\hat{\boldsymbol{n}}_1 \cdot \hat{\boldsymbol{d}}_i & -||\dot{\boldsymbol{d}}_i||\hat{\boldsymbol{n}}_2 \cdot \hat{\boldsymbol{d}}_i \end{bmatrix}\begin{bmatrix}\frac{u_1}{u_{\max}}\\ \frac{u_2}{u_{\max}}\end{bmatrix} \geq \begin{bmatrix}\dot{\boldsymbol{r}}_i \cdot \hat{\boldsymbol{r}}_i\\ -\dot{\boldsymbol{d}}_i \cdot \hat{\boldsymbol{d}}_i\end{bmatrix}$$

has no solution that also satisfies $\frac{1}{\alpha} \leq \frac{u_1}{u_{\max}} \leq 1$ and $\frac{1}{\alpha} \leq \frac{u_2}{u_{\max}} \leq 1$.

Proof. The proof of Lemma 3 is constructed by satisfying Lemmas 1 and 2 jointly. □

The existing ecologically-inspired robotics literature suggests employing slack variables to manage constraint incompatibility [23], [26]. However, it is unclear why one would add slack to the predator avoidance constraint when the premise of Lemma 2 is not satisfied. For this reason, we use Lemmas 1–3 to selectively relax the predator avoidance and swarming constraints; this implies an equivalent switching system that completely describes the behavior of each boid.

Proposition 1. Each boid $i \in \mathcal{B}$ can be modeled as a switching system with three states: 1) *Nominal*, which considers both behavioral constraints (7) and (9); 2) *Strained*, which relaxes the neighborhood constraint (7); and 3) *Evasive*, where the boid executes an evasive maneuver. Boid i transitions between these states based the premises of Lemmas 1–3 at each time; this is described by Fig. 1.

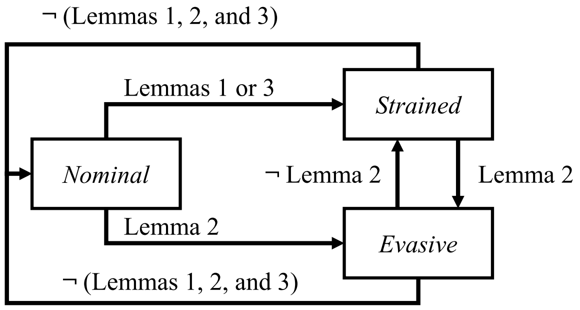

Fig. 1. A switching system that describes each boids' feasible action space based on whether the premise of Lemmas 1–3 are satisfied.

Note that defining an appropriate evasive behavior when Lemma 2 holds, e.g., a fountain [27] or flash [9] maneuver, is beyond the scope of this work; in our simulations (Section IV), we simply relax the predator-avoidance constraint. The final step is to tune the system parameters, which we discuss, along with the simulation results, in the following section.

IV. Simulation

To validate our optimal control policy, we solved Problem 1 for $N = 15$ boids over a 120 second time interval. In this section, we present our simulation parameters and the physical intuition behind them, followed by simulations that demonstrate the desired cluster flocking and predator avoidance behaviors. Additional details and simulation videos can be found on the dedicated website of manuscript, https://sites.google.com/view/ud-ids-lab/swarming.

Based on the information given in [9], we selected a diameter of 5 cm for each boid, which implies an optimal speed of approximately 12.5 cm/s. Intuitively, it is desirable for each boid $i \in \mathcal{B}$ to have a small actuation limit relative to the desired speed. Each boid ought to approach its neighborhood center $\boldsymbol{c}_i$ at a high speed, overshoot it, and circle back toward $\boldsymbol{c}_i$ in a wide arc. This circling motion will also influence the topology of the Voronoi neighborhoods, which will further perturb the flock. Ideally, these perturbations will push some boids to the edge of the flock to counteract flock collapse [7]. Additionally, we select a square domain $\mathcal{P}$ that is large enough for cluster flocking to occur. We summarize our simulation parameters in Table I.

TABLE I

Simulation parameters used to generate swarming behavior.

Domain Size (m)	v^* (m/s)	$u_{\max}$ (m/s^2)	R (cm)	Γ (cm)
6	0.125	0.1	2.5	25

To simulate the swarming behavior, we initialize all boids at rest with random initial positions within the domain $\mathcal{P}$ such that none overlap. At each time step, we solve Problem 1 and relax constraints according to Proposition 1. The behavior of the swarm is visualized in Fig. 2, which shows two time snapshots from the simulation. Figure 2 (left) shows the boids swarming toward the south-east 51 seconds into the simulation, and Fig. 2 (right) shows motion qualitatively similar to the ball behavior at 85 seconds as described by [9]–where the boids form a tight cluster and follow a circular motion.

Next, we introduce a simple predator model. The data in [9] imply that individual sand-eels treat predators as a moving obstacles. In fact, they explicitly state that "... the mackerel ate very few of the sand-eels throughout the duration of the experiment ..."—implying that the predator avoidance behavior ought to emerge without an antagonistic predator. With this justification, our predator follows a simple rule: orient toward the center of the boid flock and travel in a straight line for 8 seconds, which we found to be a reasonable tradeoff for the predator to make several passes through the boids. The predator moves 20% faster than the boids, and as such it is able to pass through the swarm and influence its behavior. A simulation snapshot is presented in Fig. 3 near $t = 52$ s, where the boids qualitatively exhibit the vacuole behavior seen in the sand-eel experiments [9].

Finally, we saved the size of each boids' neighborhood (Definition 1) at each time instant throughout the simulation. A histogram of neighborhood size is given in Fig. 4 for the simulation containing the predator. The distribution of neighborhood sizes displays positive skewness, with 4 neighbors being the most frequent. This supports existing results in the biology literature [16], which claims that only considering 3–5 neighbors may be optimal for predator avoidance in 2D swarms.

V. Conclusion

We constructed a decentralized control policy to generate emergent swarming behavior for boids operating in a constrained environment. We extended current approaches beyond control minimization and instead considered an optimal speed. We rigorously linked our event-triggered scheme for constraint relaxation to a switching system, which guarantees recursive feasibility without the use of slack variables. To verify the emergence of swarming behavior, we performed two simulations; one with no predator, and the second with a velocity obstacle that tracks the centroid of the flock.

Future work includes extensions to $\mathbb{R}^3$ with explicit collision avoidance constraints. Further exploring the distribution of neighborhood size for Voronoi neighborhoods is another compelling direction. Finally, experiments with physical robots will likely yield valuable insights.

References

[1] H. Oh, A. R. Shirazi, C. Sun, and Y. Jin, "Bio-inspired self-organising multi-robot pattern formation: A review," *Robotics and Autonomous Systems*, vol. 91, pp. 83–100, 2017.

[2] L. E. Beaver and A. A. Malikopoulos, "An Overview on Optimal Flocking," *Annual Reviews in Control*, vol. 51, pp. 88–99, 2021.

[3] I. L. Bajec and F. H. Heppner, "Organized flight in birds," *Animal Behaviour*, vol. 78, no. 4, pp. 777–789, 10 2009.

[4] H. M. La, R. Lim, and W. Sheng, "Multirobot cooperative learning for predator avoidance," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 1, pp. 52–63, 1 2015.

[5] K. Morihiro, T. Isokawa, H. Nishimura, and N. Matsui, "Emergence of Flocking Behavior Based on Reinforcement Learning," in *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*, 2006, pp. 699–706.

[6] C. Hahn, T. Phan, T. Gabor, L. Belzner, and C. Linnhoff-Popien, "Emergent Escape-based Flocking Behavior using Multi-Agent Reinforcement Learning," in *Artificial Life Conference*, 2019, pp. 598–605.

[7] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: Algorithms and theory," *IEEE Transactions on Automatic Control*, vol. 51, no. 3, pp. 401–420, 3 2006.

[8] Y. Koren and J. Borenstein, "Potential Field Methods and their Inherent Limitations for Mobile Robot Navigation," in *Proceedings of the 1991 IEEE International Conference on Robotics and Automation*, 1991.

[9] T. J. Pitcher and C. J. Wyche, "Predator-avoidance behaviours of sand-eel schools: why schools seldom split," in *Predators and Prey in Fishes*, 1983, pp. 193–204.

[10] L. E. Beaver and A. A. Malikopoulos, "Beyond Reynolds: A Constraint-Driven Approach to Cluster Flocking," in *IEEE 59th Conference on Decision and Control*, 2020, pp. 208–213.

[11] L. E. Beaver, M. Dorothy, C. Kroninger, and A. A. Malikopoulos, "Energy-Optimal Motion Planning for Agents: Barycentric Motion and Collision Avoidance Constraints," in *2021 American Control Conference*, 2021, pp. 1037–1042.

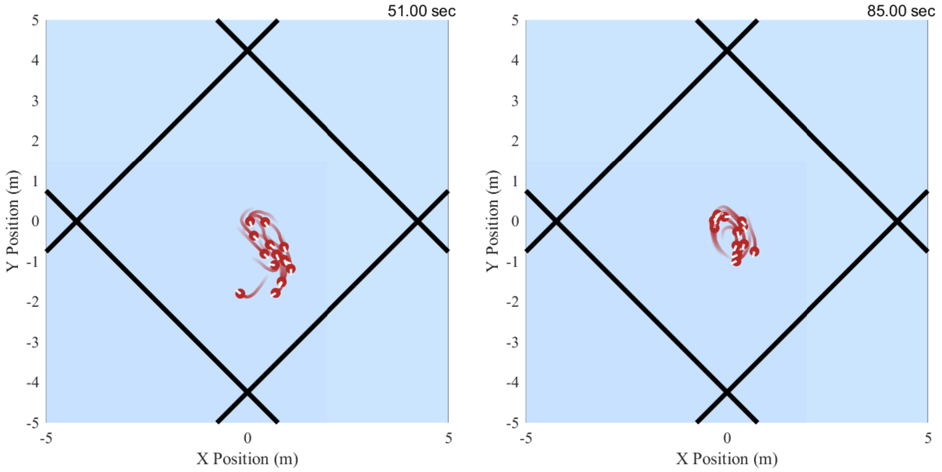

Fig. 2. Boids forming an initial flock at $t = 51$ (left) and transitioning to a ball at $t = 85$ (right) seconds; tails show 8 seconds of trajectory history.

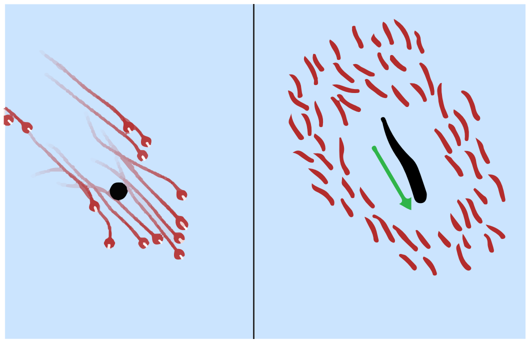

Fig. 3. Left: apparent vacuole behavior exhibited by the boids the predator approaches from behind. Right: vacuole behavior observed in sand-eels, recreated from [9].

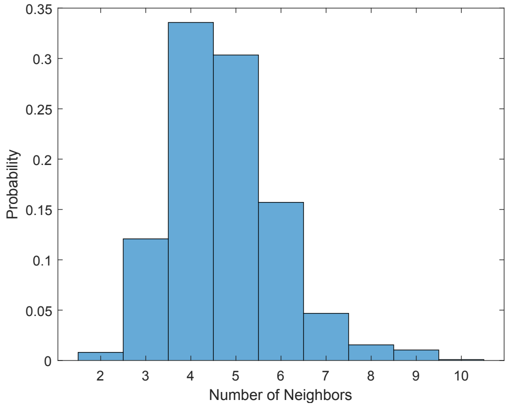

Fig. 4. Neighborhood size histogram for $N = 15$ boids during the 120 second simulation with a predator.

[12] L. E. Beaver, C. Kroninger, and A. A. Malikopoulos, "An Optimal Control Approach to Flocking," in *2020 American Control Conference*, 2020, pp. 683–688.

[13] L. Wang, A. D. Ames, and M. Egerstedt, "Safety barrier certificates for collisions-free multirobot systems," *IEEE Transactions on Robotics*, vol. 33, no. 3, pp. 661–674, 2017.

[14] B. T. Fine and D. A. Shell, "Unifying microscopic flocking motion models for virtual, robotic, and biological flock members," *Autonomous Robots*, vol. 35, no. 2-3, pp. 195–219, 10 2013.

[15] E. Cristiani, P. Frasca, and B. Piccoli, "Effects of anisotropic interactions on the structure of animal groups," *Journal of Mathematical Biology*, vol. 62, no. 4, pp. 569–588, 4 2011.

[16] M. Ballerini, N. Cabibbo, R. Candelier, A. Cavagna, E. Cisbani, I. Giardina, V. Lecomte, A. Orlandi, G. Parisi, A. Procaccini, M. Viale, and V. Zdravkovic, "Interaction ruling animal collective behavior depends on topological rather than metric distance: Evidence from a field study," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 105, no. 4, pp. 1232–1237, 2008.

[17] L. Zhou and S. Li, "Distributed model predictive control for multi-agent flocking via neighbor screening optimization," *International Journal of Robust and Nonlinear Control*, vol. 27, no. 9, pp. 1690–1705, 6 2017.

[18] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," in *2019 18th European Control Conference, ECC 2019*. Institute of Electrical and Electronics Engineers Inc., 6 2019, pp. 3420–3431.

[19] C. Hahn, F. Ritz, P. Wikidal, T. Phan, T. Gabor, and C. Linnhoff-Popien, "Foraging Swarms using Multi-Agent Reinforcement Learning," in *Artificial Life Conference*, 2020, pp. 333–340.

[20] D. Morgan, G. P. Subramanian, S.-J. Chung, and F. Y. Hadaegh, "Swarm assignment and trajectory optimization using variable-swarm, distributed auction assignment and sequential convex programming," *International Journal of Robotics Research*, vol. 35, no. 10, pp. 1261–1285, 2016.

[21] G. Notomista and M. Egerstedt, "Constraint-Driven Coordinated Control of Multi-Robot Systems," in *Proceedings of the 2019 American Control Conference*, 2019.

[22] L. E. Beaver and A. A. Malikopoulos, "Constraint-driven optimal control of multi-agent systems: A highway platooning case study," *IEEE Control Systems Letters*, vol. 6, pp. 1754–1759, 2022.

[23] M. Egerstedt, J. N. Pauli, G. Notomista, and S. Hutchinson, "Robot ecology: Constraint-based control design for long duration autonomy," *Annual Reviews in Control*, vol. 46, pp. 1–7, 1 2018.

[24] K. Xu, W. Xiao, and C. G. Cassandras, "Feasibility Guaranteed Traffic Merging Control Using Control Barrier Functions," in *2022 American Control Conference*, 2022, pp. 2039–2314.

[25] Y. V. Pant, H. Abbas, and R. Mangharam, "Smooth Operator: Control using the Smooth Robustness of Temporal Logic," in *Proceedings of the 2017 IEEE Conference on Control Technology and Applications*, 2017, pp. 1235–1240.

[26] T. Ibuki, S. Wilson, J. Yamauchi, M. Fujita, and M. Egerstedt, "Optimization-Based Distributed Flocking Control for Multiple Rigid Bodies," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1891–1898, 4 2020.

[27] F. Berlinger, P. Wulkop, and R. Nagpal, "Self-Organized Evasive Fountain Maneuvers with a Bioinspired Underwater Robot Collective," in *2021 IEEE International Conference on Robotics and Automation*. Institute of Electrical and Electronics Engineers (IEEE), 10 2021, pp. 9204–9211.