PPO-driven Swarm Control: A Hybrid Multi-Robot Framework Combining Consensus, Potential Fields, and CRN-Based Role Switching

Ayushman Mishra
aymisxx@proton.me

***Abstract*—This project presents a hybrid control framework for distributed multi-robot coverage that integrates reinforcement learning with established principles from Multi-Robot Systems (MRS). The application focuses on vegetation-driven exploration in precision agriculture, where a team of unmanned aerial vehicles must efficiently traverse and assess a field represented by a normalized vegetation index derived from satellite imagery. The proposed system combines a microscopic controller learned through Proximal Policy Optimization (PPO) with macroscopic coordination layers grounded in potential-field methods, graph-based consensus, and stochastic Chemical Reaction Network (CRN) like role-switching dynamics.**

The mathematical model describes each robot as a single-integrator agent equipped with local sensing and bounded communication. Attractive and repulsive potential fields regulate movement with respect to environmental gradients and inter-agent separation, while consensus terms promote neighborhood agreement on coverage-related quantities. Role-switching is modeled using CRN-inspired stochastic processes that allow agents to alternate between explorer, surveyor, defender, and idle modes in response to local density and NDVI variations. The analysis focuses on coverage efficiency, dispersion, collision avoidance, consensus convergence, and the stability properties induced by the hybrid controller.

Simulation results validate the theoretical expectations and demonstrate clear performance gains as each coordination layer is added. PPO-only swarms exhibit partial exploration but suffer from clustering and redundant trajectories. Incorporating potential fields improves dispersion, while consensus dynamics further reduce coverage imbalance across the field. The full hybrid controller with CRN-based roles achieves the greatest spatial distribution, lowest redundancy, and strongest alignment with NDVI gradients. Trajectory visualizations, coverage maps, consensus-error trends, and role-distribution plots collectively show that the hybrid system produces coherent, scalable, and robust multi-robot behavior.

Overall, the project illustrates how machine learning and classical MRS theory can be fused to achieve reliable and adaptive swarm-level coverage in complex environments.

I. Introduction

Multi-robot systems are increasingly used in environments where large areas must be explored, monitored, or serviced without relying on centralized control. When individual robots operate with limited sensing and communication, their coordinated behavior emerges from decentralized rules that shape how they move, share information, and respond to local conditions. Applications such as agricultural surveying, environmental mapping, inspection, and search tasks benefit from these distributed strategies, since they naturally scale with the number of robots and remain functional even when individual agents fail or lose communication.

Classical multi-robot coordination methods offer several well-established tools for building such systems. Potential-field navigation influences robot motion through attractive and repulsive forces, consensus dynamics help a group align or share information consistently, and stochastic task allocation models describe how robots change roles in response to local measurements. These methods are mathematically grounded and come with strong guarantees on stability, convergence, and safety.

In parallel, reinforcement learning has become popular for synthesizing control policies in complex and uncertain environments. Although learning-based controllers can adapt to rich sensory input, they often lack an inherent notion of multi-agent structure and may produce behaviors that do not scale well to larger teams.

This project explores a hybrid approach that brings these two perspectives together. A PPO-based learned controller provides each robot with a local navigation policy informed by a vegetation index field, while additional coordination layers introduce potential-field interactions, consensus terms, and role switching inspired by chemical reaction models. The goal is to show how learning and analytical methods can complement each other to produce reliable, scalable, and well-organized swarm behavior.

II. Mathematical Model

This section establishes the mathematical foundations of the proposed PPO-driven multi-robot coverage system. The formulation integrates classical multi-robot control concepts; single-integrator kinematics, artificial potential fields, graph-based consensus, and stochastic role switching; grounded in prior work on distributed coverage [1], multi-UAV area coverage [2], chemotaxis-inspired swarm motion [3], and ensemble-level stochastic task allocation [4].

We consider a team of N UAVs deployed in a bounded planar domain $\Omega \subset \mathbb{R}^2$, representing an agricultural field containing a scalar vegetation-density function.

A. Assumptions and Constraints

The model is built upon the following assumptions, consistent with decentralized coverage literature [1]

- Fully actuated planar UAVs modeled as single-integrator agents.
- Fixed flight altitude, reducing motion to 2D.
- Local scalar-field sensing only.
- Limited-range communication inducing a time-varying graph.
- Collision avoidance enforced through repulsive potentials.
- Coverage objective emphasizing first-visit reward collection.

B. *Robot Dynamics and Environmental Field*

1) Single-Integrator Robot Model: The kinematic model for robot i is

$$\dot{x}_i(t) = u_i(t), \tag{1}$$

where $x_i(t) \in \Omega$ is the position and $u_i(t)$ is the velocity command. *Explanation* The equation states that the UAV's instantaneous velocity is directly controlled by the control input. This abstraction is widely used in formation and coverage algorithms.

2) Vegetation Scalar Field: Let the normalized vegetation index be

$$\phi : \Omega \to [0, 1]. \tag{2}$$

Explanation $\phi(p)$ assigns a vegetation richness to each point p. Higher values indicate more fertile areas needing more surveillance [2].

We track first-touch coverage using the binary indicator

$$C(p,t) = \begin{cases} 1, & \text{if some robot has visited } p \text{ by time } t, \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

Explanation $C(p,t) = 1$ marks that location p no longer yields reward if revisited later.

The total vegetation collected by time T is

$$J(T) = \int_{\Omega} \phi(p)\, C(p,T)\, \mathrm{d}p. \tag{4}$$

Explanation Only the first visit contributes reward, weighted by the local vegetation index.

3) Local Observation Model: Each robot sees only a local window

$$o_i(t,\xi) = \phi\big(x_i(t) + \xi\big), \quad \xi \in W. \tag{5}$$

Explanation The set $W \subset \mathbb{R}^2$ denotes a fixed local observation window centered at the robot position, corresponding to the 128×128 patch used for PPO input.

C. *Hybrid Control Architecture*

Each robot's control input consists of

$$u_i(t) = w_{\mathrm{rl}}(r_i) u_i^{\mathrm{PPO}} + w_{\mathrm{pf}}(r_i) u_i^{\mathrm{PF}} + w_{\mathrm{cons}}(r_i) u_i^{\mathrm{cons}},$$

with each term described below.

1) PPO-Based Controller: The learned controller outputs

$$u_i^{\mathrm{PPO}}(t) = f_\theta\big(o_i(t), z_i(t)\big). \tag{6}$$

Explanation The PPO network maps the local NDVI patch and internal features to a motion direction, imitating single-agent coverage performance [5].

2) Artificial Potential Field: Define the potential

$$\begin{aligned} U_i(x) = & -\alpha\, \phi(x) + \beta \sum_{j \neq i} \psi\left(x - x_j(t)\right) \\ & + \gamma\, \eta\left(C(x,t)\right). \end{aligned} \tag{7}$$

Explanation

- The first term attracts robots to high-NDVI regions.
- The second term repels nearby robots for collision avoidance.
- The third term discourages revisiting previously covered cells.

The potential-field control is

$$u_i^{\mathrm{PF}}(t) = -\nabla_x U_i(x)\Big|_{x = x_i(t)}. \tag{8}$$

Explanation Robots follow descending gradients of U_i, similar to chemotaxis-inspired swarm motion [3]. The repulsive control term u_i^{rep} analyzed in Section III is contained within u_i^{PF} through the gradient of the inter-agent repulsion component of $U_i(x)$.

3) Consensus Control Layer: Let $\mathcal{N}_i(t)$ be robot i's neighbor set. The alignment term is

$$u_i^{\mathrm{cons}}(t) = -k_{\mathrm{cons}} \sum_{j \in \mathcal{N}_i(t)} a_{ij}(t)\, \big(x_i(t) - x_j(t)\big). \tag{9}$$

Explanation This reduces excessive dispersion and keeps the swarm loosely cohesive, reflecting standard distributed consensus behavior [1].

D. *Role-Dependent Hybrid Control*

Each robot takes a role $r_i(t) \in \{\mathrm{E}, \mathrm{S}, \mathrm{D}, \mathrm{I}\}$. Weights modulate behavior

$$\text{Explorer} \quad w_{\mathrm{rl}} \gg w_{\mathrm{pf}}, w_{\mathrm{cons}}, \tag{10}$$

$$\text{Defender} \quad w_{\mathrm{pf}}, w_{\mathrm{cons}} \gg w_{\mathrm{rl}}. \tag{11}$$

Explanation Explorers rely primarily on PPO-driven exploration, while defenders emphasize cohesion and collision avoidance.

Substituting all components gives the final dynamics

$$\begin{aligned} \dot{x}_i(t) = & \, w_{\mathrm{rl}}(r_i) f_\theta(o_i, z_i) \\ & - w_{\mathrm{pf}}(r_i)\, \nabla_x U_i(x_i(t)) \\ & - w_{\mathrm{cons}}(r_i)\, k_{\mathrm{cons}} \sum_{j \in \mathcal{N}_i(t)} a_{ij}(t)(x_i - x_j). \end{aligned} \tag{12}$$

Explanation This hybrid law smoothly blends learned behavior with structured swarm control primitives.

E. *CRN-Inspired Stochastic Role Switching*

Motivated by ensemble task allocation [4], role updates follow a Markov process

$$\mathbb{P}[r_i(t + \Delta t) = r' \mid r_i(t) = r,\, \xi_i(t)] = P_{rr'}(\xi_i(t)). \tag{13}$$

Explanation Transition probabilities depend on local metrics such as coverage density or effective speed.

The mean-field population model is

$$\dot{\rho}_r(t) = \sum_s \rho_s(t)\lambda_{sr}(\bar{\xi}) - \rho_r(t)\sum_s \lambda_{rs}(\bar{\xi}), \tag{14}$$

Explanation This describes the expected fraction of robots in each role, analogous to species concentrations in a chemical reaction network.

F. Robot Dynamics and Environmental Field

1) Single-Integrator Dynamics: Each robot i has planar position $x_i(t) \in \Omega$ and obeys

$$\dot{x}_i(t) = u_i(t), \tag{15}$$

where $u_i(t)$ is the hybrid control input defined later.

2) Vegetation Scalar Field: Let

$$\phi : \Omega \to [0, 1], \tag{16}$$

represent the normalized vegetation index. This structure aligns with UAV coverage literature where robots respond to scalar fields [2].

We define a binary coverage state

$$C(p,t) = \begin{cases} 1, & \text{if any robot has visited } p \text{ by time } t, \\ 0, & \text{otherwise.} \end{cases} \tag{17}$$

Total collected vegetation reward is

$$J(T) = \int_\Omega \phi(p)\, C(p,T)\, dp. \tag{18}$$

3) Local Observation Model: Each UAV receives a local NDVI window

$$o_i(t,\cdot) = \phi(x_i(t) + \cdot)\Big|_W, \tag{19}$$

consistent with distributed sensing assumptions in [1].

G. Hybrid Control Architecture

Each robot's control input is composed of

1) PPO-based learned motion term u_i^{PPO},
2) Potential-field term u_i^{PF} for attraction, repulsion, and revisit-avoidance,
3) Consensus term u_i^{cons} promoting cohesion,
4) Role-dependent weighting via CRN-based switching.

1) PPO-based Local Controller: The PPO policy is represented as

$$u_i^{\mathrm{PPO}}(t) = f_\theta(o_i(t), z_i(t)), \tag{20}$$

analogous to learned local controllers increasingly used in multi-UAV coverage planning [5].

2) Artificial Potential Fields: Following potential-field coverage and chemotaxis-inspired models [3], define

$$U_i(x) = -\alpha\phi(x) + \beta\sum_{j\neq i}\psi(x - x_j) + \gamma\eta(C(x,t)). \tag{21}$$

Control

$$u_i^{\mathrm{PF}}(t) = -\nabla_x U_i(x)\big|_{x=x_i(t)}. \tag{22}$$

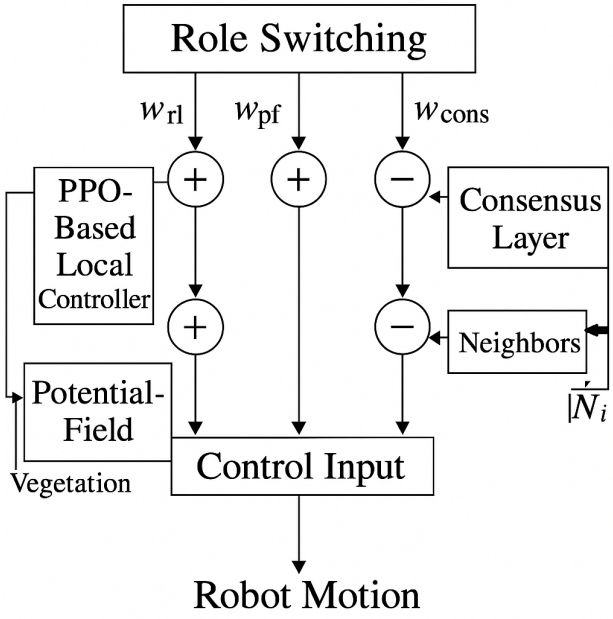

Fig. 1: Hybrid control architecture showing the PPO-based local controller, potential-field module, consensus layer, and CRN-based role-switching mechanism.

3) Consensus Term: Let $\mathcal{G}(t)$ be the communication graph. The consensus alignment is

$$u_i^{\mathrm{cons}}(t) = -k_{\mathrm{cons}}\sum_{j\in\mathcal{N}_i(t)} a_{ij}(t)(x_i(t) - x_j(t)), \tag{23}$$

consistent with established consensus schemes used in distributed robotic coverage [1].

H. Role-Dependent Hybrid Control Law

Each robot has role $r_i \in \{\mathrm{E}, \mathrm{S}, \mathrm{D}, \mathrm{I}\}$.

Hybrid control

$$u_i(t) = w_{\mathrm{rl}}(r_i)\, u_i^{\mathrm{PPO}}(t) + w_{\mathrm{pf}}(r_i)\, u_i^{\mathrm{PF}}(t) + w_{\mathrm{cons}}(r_i)\, u_i^{\mathrm{cons}}(t). \tag{24}$$

I. CRN-Inspired Stochastic Role Switching

Motivated by ensemble-level stochastic task allocation [4], let $\xi_i(t)$ denote a vector of locally measurable quantities available to robot i, such as local NDVI statistics, neighborhood density, or recent motion magnitude. Each robot updates its role according to

$$\mathbb{P}[r_i(t+\Delta t) = r' \mid r_i(t) = r,\ \xi_i(t)] = P_{rr'}(\xi_i(t)). \tag{25}$$

At the swarm level, expected role fractions satisfy the mean-field ODE

$$\dot{\rho}_r(t) = \sum_s \rho_s(t)\lambda_{sr}(\bar{\xi}) - \rho_r(t)\sum_s \lambda_{rs}(\bar{\xi}), \tag{26}$$

similar to ensemble control and CRN-inspired modeling in [4].

J. Model Variables and Parameters

This subsection summarizes all variables, state quantities, and parameters used in the mathematical model. These definitions follow standard notation in multi-robot coordination and match the structure of Fig. 2.

1) Robot States and Geometry:

- $x_i(t) \in \Omega \subset \mathbb{R}^2$ — position of robot i at time t.
- $u_i(t) \in \mathbb{R}^2$ — control input (velocity command).
- $d_{ij}(t) = x_i(t) - x_j(t)$ — inter-robot distance.
- $\mathcal{N}_i(t)$ — communicationinteraction neighbor set of robot i.
- $\mathcal{N}_i(t)$ — number of neighbors of robot i.
- R_{sense} — sensing radius for local NDVI window.
- R_{comm} — communicationconsensus range.

2) Environmental Field and Coverage Quantities:

- $\phi(p) \in [0, 1]$ — normalized vegetation index (NDVI) at location p.
- $o_i(t)$ — local NDVI observation patch extracted around robot i.
- $C(p, t) \in \{0, 1\}$ — binary visited indicator for first-visit reward.
- $J(T)$ — total accumulated vegetation reward up to time T.

3) Control Architecture Variables:

- u_i^{PPO} — control vector generated by the PPO policy.
- u_i^{PF} — potential-field control vector.
- u_i^{cons} — consensus alignment control.
- $U_i(x)$ — potential function combining attraction, repulsion, and revisit penalty.
- k_{cons} — consensus gain.
- α, β, γ — PF weights for attraction, repulsion, and revisit-avoidance.

4) Role Switching Parameters:

- $r_i(t) \in \{\text{E}, \text{S}, \text{D}, \text{I}\}$ — robot role (Explorer, Surveyor, Defender, Idle).
- $w_{\text{rl}}, w_{\text{pf}}, w_{\text{cons}}$ — role-dependent weights.
- $P_{rr'}(\xi_i)$ — transition probability from role r to r'.
- λ_{rs} — macroscopic transition rates in the mean-field CRN model.
- $\rho_r(t)$ — fraction of robots in role r at time t.

5) Geometric Model Figure (Placeholder): Figure 2 illustrates

- robot position x_i in a global coordinate frame,
- sensing radius R_{sense},
- communication radius R_{comm},
- inter-robot distance d_{ij},
- local NDVI window used for PPO,
- neighbor set $\mathcal{N}_i$.

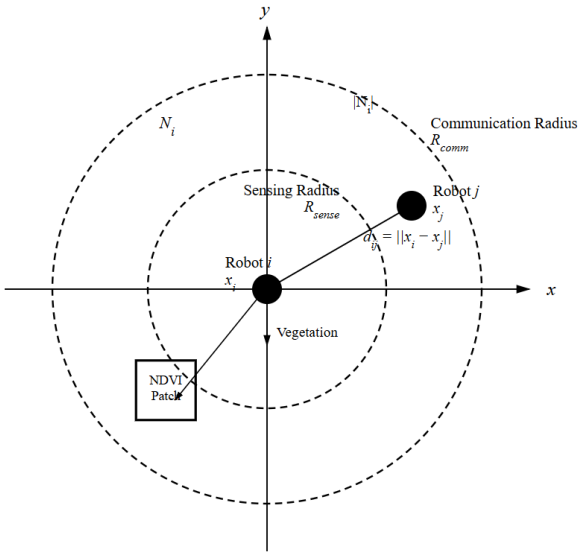

Fig. 2: Geometric relationships and variables used in the multi-robot model.

III. Theoretical Analysis

In this section, we establish formal guarantees for three key properties of the proposed multi-robot model (1) consensus convergence, (2) collision avoidance via repulsive potentials, and (3) boundary invariance and boundedness. We also demonstrate how the hybrid controller achieves safe, cohesive, and NDVI-seeking collective behavior. Our analysis draws from classical consensus theory [7], [11], potential-field navigation [8], [9], and Lyapunov-based invariance arguments [10].

A. Preliminaries

Each robot maintains a planar position vector

$$p_i \in \mathbb{R}^2, \qquad i = 1, \ldots, N, \tag{27}$$

which we stack into a global state vector

$$p = \begin{bmatrix} p_1^T & p_2^T & \cdots & p_N^T \end{bmatrix}^T \in \mathbb{R}^{2N}. \tag{28}$$

This representation allows the team's configuration to be analyzed using matrix tools such as graph Laplacians.

The communication network is modeled as

$$\mathcal{G} = (\mathcal{V}, \mathcal{E}), \tag{29}$$

where the edges describe available inter-robot communication links. The corresponding Laplacian L encodes connectivity.

Each robot applies the hybrid control law

$$u_i = w_{\text{rl}}(r_i)\, u_i^{\text{PPO}} + w_{\text{cons}}(r_i)\, u_i^{\text{cons}} + w_{\text{pf}}(r_i)\, u_i^{\text{PF}}, \tag{30}$$

where the weights form a convex combination and reflect the currently assigned role.

The discrete-time dynamics are

$$p_i(t+1) = p_i(t) + u_i(t), \tag{31}$$

which propagate the robot positions forward at each control cycle.

B. *Property 1 - Consensus Convergence*

The consensus contribution to the control input is

$$u_i^{\text{cons}} = k_{\text{cons}} \sum_{j \in \mathcal{N}_i} (p_j - p_i). \tag{32}$$

This term pulls each robot toward the average position of its neighbors, promoting cohesion.

Stacking all agents yields the global closed-loop dynamics

$$p(t+1) = p(t) - k_{\text{cons}}(L \otimes I_2)\, p(t), \tag{33}$$

a standard linear consensus system [11].

Define the team centroid

$$\bar{p}(t) = \frac{1}{N} \sum_{i=1}^{N} p_i(t), \tag{34}$$

and the consensus error

$$E(t) = \sum_{i=1}^{N} \|p_i(t) - \bar{p}(t)\|^2. \tag{35}$$

For a connected graph, Laplacian eigenvalues satisfy

$$0 = \lambda_1 < \lambda_2 \leq \cdots \leq \lambda_N, \tag{36}$$

where λ_2 (algebraic connectivity) governs convergence speed.

Classical spectral analysis [7] yields

$$E(t+1) \leq \left(1 - 2k_{\text{cons}}\lambda_2 + k_{\text{cons}}^2 \lambda_N^2\right) E(t). \tag{37}$$

Thus, if

$$0 < k_{\text{cons}} < \frac{2}{\lambda_N}, \tag{38}$$

the multiplicative factor is strictly less than one, ensuring

$$\lim_{t \to \infty} E(t) = 0. \tag{39}$$

Hence, the robots reach consensus asymptotically.

C. *Property 2 - Collision Avoidance*

The inter-robot distance is

$$d_{ij} = \|p_i - p_j\|. \tag{40}$$

Repulsion activates when robots come within a safety radius.

The repulsive control is

$$u_i^{\text{rep}} = \sum_{j \neq i} k_{\text{rep}} \left(\frac{1}{d_{ij}} - \frac{1}{R_{\text{rep}}}\right) \frac{p_i - p_j}{d_{ij}}, \tag{41}$$

which is derived from a classical artificial potential [9]

$$\phi(d_{ij}) = \frac{1}{2} k_{\text{rep}} \left(\frac{1}{d_{ij}} - \frac{1}{R_{\text{rep}}}\right)^2. \tag{42}$$

The total repulsive potential is

$$V_{\text{rep}}(p) = \sum_{ij} \phi(d_{ij}). \tag{43}$$

Since $\phi(d_{ij}) \to \infty$ as $d_{ij} \to 0^+$, robots cannot collide while applying bounded controls. Thus, the safe set

$$\mathcal{S} = \{\, p \mid d_{ij} \geq d_{min} > 0,\ \forall i, j \,\} \tag{44}$$

is positively invariant.

More formally, $V_{\text{rep}}(p)$ acts as a barrier function: as any pairwise distance approaches zero, $V_{\text{rep}}(p)$ becomes unbounded, and the repulsive feedback in (41) generates an outward velocity component along the line connecting the agents. Therefore, trajectories initialized with $d_{ij}(0) \geq d_{\min}$ cannot cross the boundary $d_{ij} = d_{\min}$ under the closed-loop dynamics, since doing so would require $V_{\text{rep}}(p)$ to increase without bound in finite time. Hence, for all $t \geq 0$, we maintain $d_{ij}(t) \geq d_{\min}$, implying collision avoidance and positive invariance of $\mathcal{S}$.

D. *Property 3 - Boundary Invariance and Boundedness*

Boundary invariance refers to the property that robot trajectories remain inside the rectangular domain for all time once initialized within it, while boundedness guarantees that all robot positions and inter-robot distances remain finite under the closed-loop dynamics.

Let the domain be

$$[y_{min}, y_{max}] \times [x_{min}, x_{max}]. \tag{45}$$

A lower-wall barrier potential is [8]

$$V_{y,\text{low}}(y_i) = \frac{1}{2} k_{\text{bnd}} \left(\frac{1}{y_i - y_{min}} - \frac{1}{\delta}\right)^2. \tag{46}$$

The complete safety potential is

$$V_{\text{safe}}(p) = V_{\text{rep}}(p) + V_{y,\text{low}} + V_{y,\text{high}} + V_{x,\text{left}} + V_{x,\text{right}}. \tag{47}$$

Since $V_{\text{safe}} \to \infty$ near walls and collisions, the sublevel sets

$$\Omega_c = \{\, p \mid V_{\text{safe}}(p) \leq c \,\} \tag{48}$$

are compact and invariant by LaSalle's principle [10].

More precisely, $V_{\text{safe}}(p)$ serves as a barrier-type energy function: as any robot approaches a boundary (e.g., $y_i \to y_{\min}$) the corresponding wall potential term (e.g., (46)) becomes unbounded, and similarly $V_{\text{rep}}(p) \to \infty$ as any $d_{ij} \to 0^+$. Therefore, if the system is initialized with finite $V_{\text{safe}}(p(0)) \leq c$, trajectories cannot reach the walls or collision configurations without causing V_{safe} to blow up. Invariance of the compact sublevel set Ω_c implies that all robot positions remain inside the bounded domain for all time, establishing boundary invariance, and that the overall state $p(t)$ remains bounded. Hence, the closed-loop trajectories satisfy $p(t) \in \Omega_c \subset \Omega$ for all $t \geq 0$.

E. *Hybrid Controller and Collective Behavior*

The full hybrid control is

$$u_i = w_{\text{rl}}(r_i)\, u_i^{\text{PPO}} + w_{\text{cons}}(r_i)\, u_i^{\text{cons}} + w_{\text{pf}}(r_i)\, u_i^{\text{PF}}, \tag{49}$$

a convex combination of fields with known stability guarantees [11].

Consensus maintains cohesion, potential fields enforce spacing and boundary safety, and PPO terms drive robots toward informative NDVI-rich regions.

Thus, the hybrid controller produces collision-free, cohesive, and environment-aware collective motion.

IV. Validation in Simulations

This section validates the proposed multi-robot model and hybrid controller using the Python simulation notebook. Our aims are to demonstrate that the simulated behavior matches the properties proven in Section III, namely: (1) consensus convergence, (2) collision avoidance and positive minimum spacing, (3) boundary invariance and boundedness, and (4) NDVI-aware coverage of the environment. We progressively increase complexity from potential-field-only motion to full hybrid control with role switching and hyperparameter sweeps.

A. NDVI Field and Swarm Initialization

The environment is a VARI-based NDVI proxy map normalized to $[0, 1]$, which provides vegetation gradients for the attraction component of the potential field. Figure 3 shows the underlying vegetation map used in all experiments.

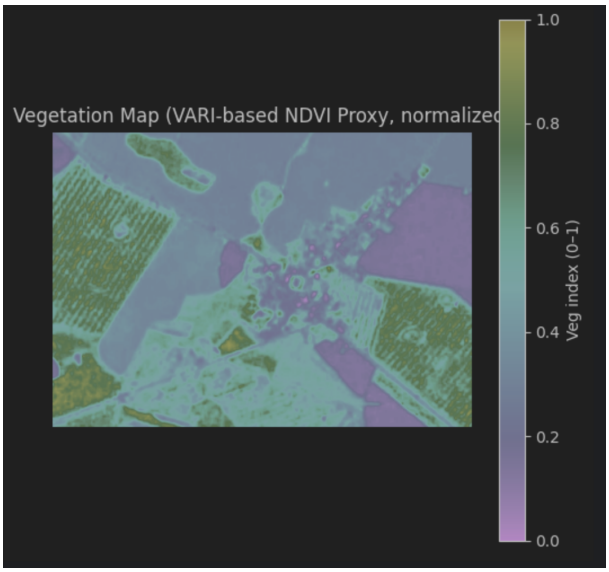

Fig. 3: Vegetation map (VARI-based NDVI proxy, normalized to $[0, 1]$).

Robots are initialized near the center of the field in a compact but slightly spread configuration (Fig. 4). The communication graph based on the proximity radius R_{comm} is shown in Fig. 5.

B. Single-Step Consensus and Potential-Field Vector Fields

To visually validate the structure of the vector fields in our model, we first compute single-step velocities for the consensus and potential-field components with the robots frozen at their initial positions.

Figure 6 shows the consensus velocity field u_i^{cons}. The arrows consistently point toward the swarm centroid, in agreement with the contraction property proven in Section III.

Figure 7 shows the potential-field velocity field u_i^{PF}, which combines repulsion, boundary forces and NDVI attraction. The vectors point away from close neighbors, inward from the map boundaries, and along vegetation gradients.

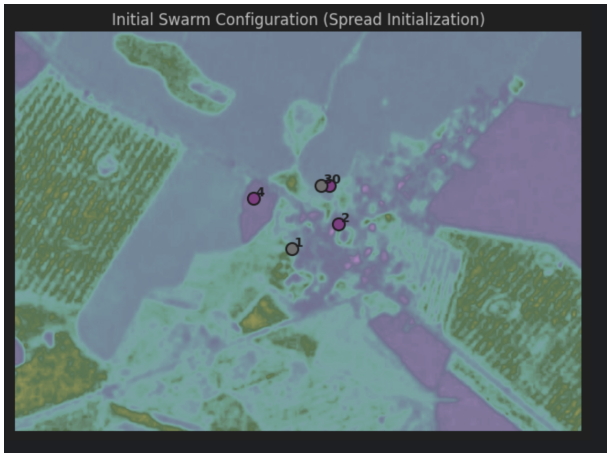

Fig. 4: Initial swarm configuration over the NDVI field.

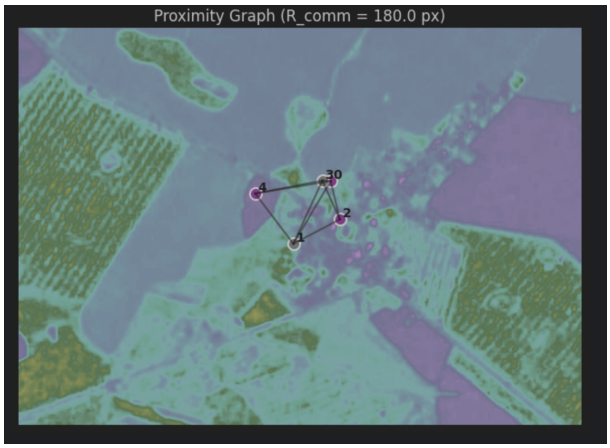

Fig. 5: Proximity graph showing communication links under R_{comm}.

Fig. 6: Consensus velocity field for a single time step (u_i^{cons}).

A simplified Python function implementing this hybrid control law in the notebook is shown in Listing 1.

Listing 1: Hybrid control step used in the simulations (simplified).

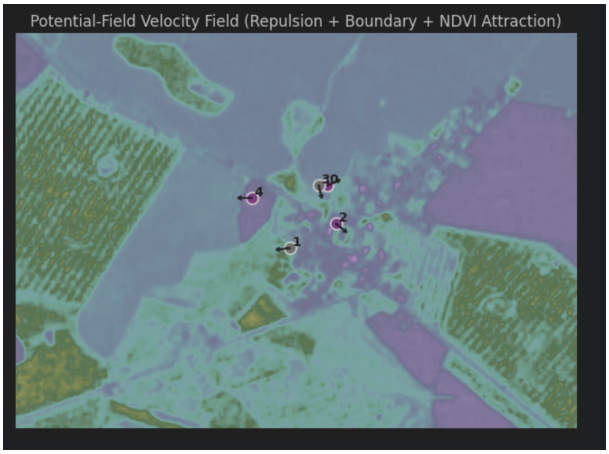

Fig. 7: Potential-field velocity field (u_i^{PF}) for a single time step.

```python
def hybrid_control_step(state, roles, params):
    """Compute one-step hybrid control for all drones."""
    # consensus, potential-field and PPO actions
    u_cons = consensus_field(state.positions, params["k_cons"], params["R_comm"])
    u_pf   = potential_field(state.positions,
                             ndvi_map=state.ndvi,
                             k_rep=params["k_rep"],
                             k_att=params["k_att"],
                             R_rep=params["R_rep"])
    u_ppo  = ppo_policy(state.obs)   # pre-trained PPO policy

    # role-dependent weights
    w_rl, w_cons, w_pf = role_weights(roles, params)

    # convex combination
    u = w_rl[:, None]*u_ppo + w_cons[:, None]*u_cons + w_pf[:, None]*u_pf
    next_positions = state.positions + params["dt"]*u
    return next_positions, u, {"u_cons": u_cons, "u_pf": u_pf, "u_ppo": u_ppo}
```

C. Potential-Field-Only Swarm Behavior

We first simulate the swarm under *potential-field-only* motion (without consensus or PPO) to isolate the effects of repulsion, boundary forces and NDVI attraction.

Figure 8 shows the drone trajectories under this controller. Agents spread out from the initial cluster while remaining within the map and drifting toward vegetation-rich regions.

To quantify the strength of the potential-field forces, we track the mean norm of u_i^{PF} over time, shown in Fig. 9. After an initial transient, the magnitude stabilizes around a constant value, indicating that the repulsive and boundary interactions remain active but bounded.

Swarm dispersion under potential fields only is illustrated in Fig. 10. The dispersion $D(t) = \sum_i \|p_i(t) - \bar{p}(t)\|^2$ initially increases as robots repel each other, then slowly decreases and stabilizes. At all times $D(t) > 0$, confirming that the swarm does not collapse to a point.

Fig. 8: Drone trajectories under potential-field-only motion.

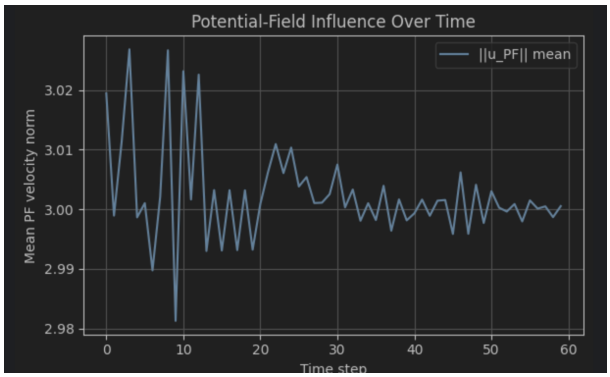

Fig. 9: Mean potential-field velocity norm over time.

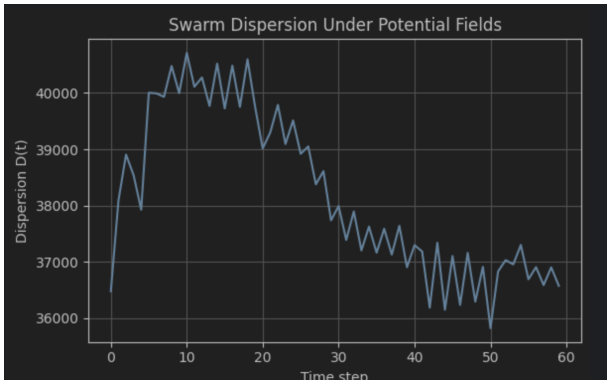

Fig. 10: Swarm dispersion under potential-field-only motion.

Finally, Fig. 11 shows the coverage ratio

$$C(t) = \frac{\text{number of unique visited pixels}}{\text{total number of pixels}}$$

as a function of time. The monotonic increase of $C(t)$ indicates that the swarm provides nontrivial coverage of the NDVI field even with potential fields alone.

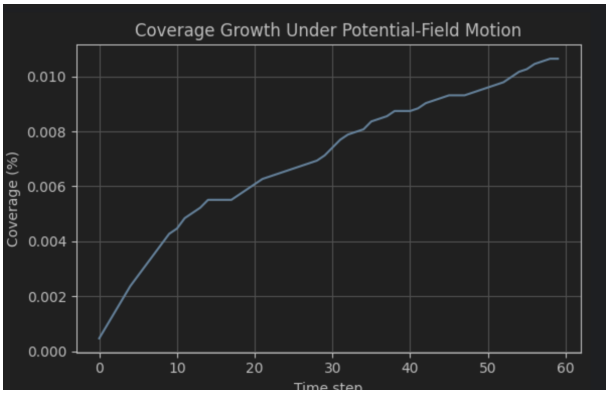

Fig. 11: Coverage growth under potential-field-only motion.

D. Hybrid Control Without Role Switching

We next activate the full hybrid controller

$$u_i = w_{\mathrm{rl}}(r_i)u_i^{\mathrm{PPO}} + w_{\mathrm{cons}}(r_i)u_i^{\mathrm{cons}} + w_{\mathrm{pf}}(r_i)u_i^{\mathrm{PF}},$$

with fixed roles and weights, to study the interaction of PPO, consensus and potential fields.

Figure 12 depicts the drone trajectories under hybrid control. Compared to the potential-field-only case, the paths are more cohesive and less divergent, reflecting the influence of consensus and PPO actions.

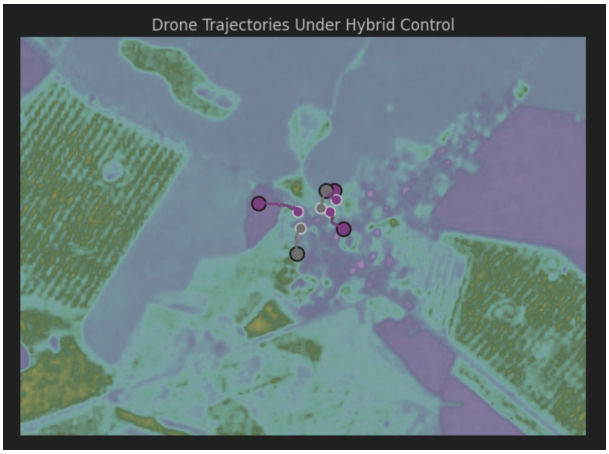

Fig. 12: Drone trajectories under hybrid control (fixed roles).

Figure 13 shows the raw mean norms $\|u^{\mathrm{PPO}}\|$, $\|u^{\mathrm{cons}}\|$ and $\|u^{\mathrm{PF}}\|$ over time. The consensus component decays as the swarm contracts, while PPO and potential-field magnitudes remain relatively steady.

To capture the effect of role-dependent weighting, Fig. 14 plots the mean norms of $w_{\mathrm{rl}}u^{\mathrm{PPO}}$, $w_{\mathrm{cons}}u^{\mathrm{cons}}$, and $w_{\mathrm{pf}}u^{\mathrm{PF}}$. The potential-field contribution remains dominant, while consensus gradually decreases as the formation stabilizes, consistent with our controller design.

Swarm dispersion under hybrid control is shown in Fig. 15. Unlike the potential-field-only case, $D(t)$ decreases sharply and then slowly converges, indicating that consensus successfully contracts the swarm while repulsion prevents collapse.

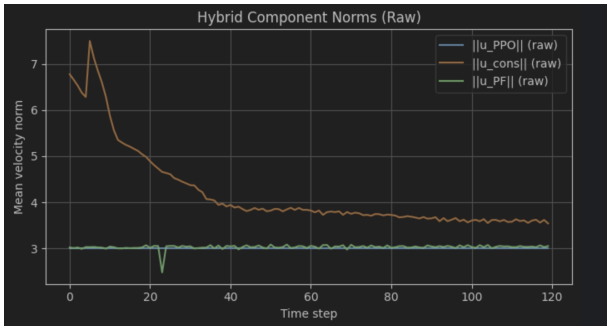

Fig. 13: Raw mean velocity norms of PPO, consensus and potential-field components under hybrid control.

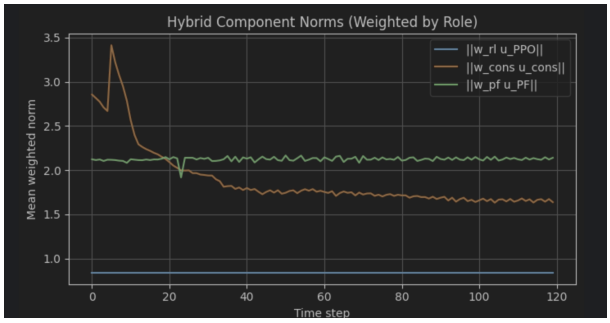

Fig. 14: Weighted mean norms of the three hybrid components under fixed-role hybrid control.

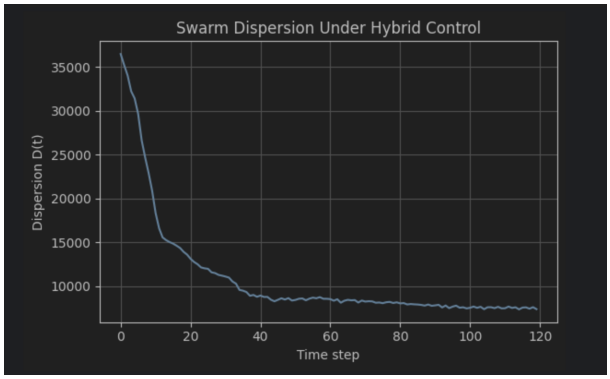

Fig. 15: Swarm dispersion under hybrid control (fixed roles).

Hybrid-control coverage performance is shown in Fig. 16. Coverage grows faster and to a higher value than in the potential-field-only experiment, highlighting the additional exploration capability provided by the PPO-based term.

E. Role-Switching Hybrid Control

We then enable the role-switching mechanism so that robots can dynamically change between *explorer*, *surveillor*, *defender* and *idle* states based on local conditions. A simplified version of the role-update logic used in the notebook is given in Listing 2.

Listing 2: Role-switching logic (simplified).

```
def update_roles(state, roles, params):
```

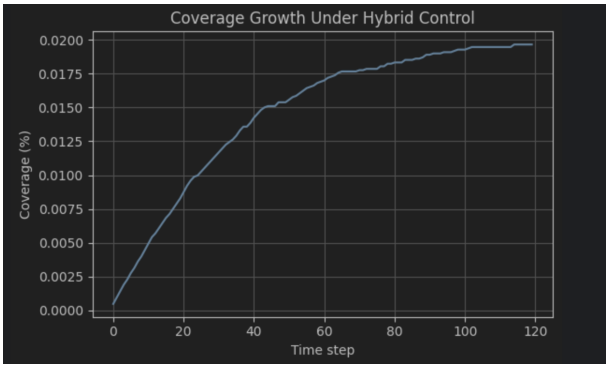

Fig. 16: Coverage growth under hybrid control (fixed roles).

```
    """Heuristic role switching based on local NDVI
        and dispersion."""
    new_roles = roles.copy()
    for i, pos in enumerate(state.positions):
        local_ndvi = read_ndvi_patch(state.ndvi, pos
            )
        if local_ndvi.mean() > params["ndvi_high"]:
            new_roles[i] = "explorer"
        elif state.dispersion < params["disp_low"]:
            new_roles[i] = "defender"
        elif state.dispersion > params["disp_high"]:
            new_roles[i] = "surveillor"
        else:
            new_roles[i] = "idle"
    return new_roles
```

Figure 17 plots the number of robots in each role for the initial role-switching configuration. Defenders dominate during most of the run, while explorers are intermittently activated, and idle robots quickly transition into active roles. Defender robots primarily serve a stabilizing function in the swarm. By assigning higher weights to the potential-field and consensus terms, defenders maintain inter-agent spacing, prevent boundary accumulation, and preserve swarm cohesion while exploration is carried out by other agents. This role allows the swarm to remain collision-free and well-distributed, acting as a structural backbone that supports exploration without central coordination.

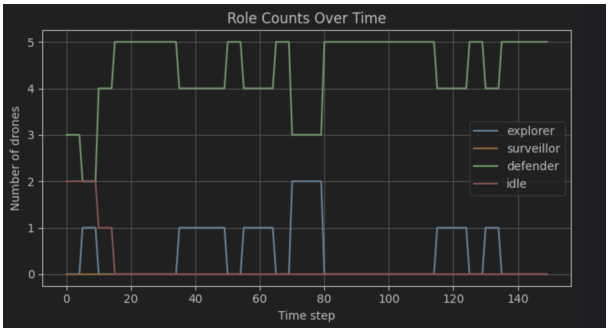

Fig. 17: Role counts over time under the initial role-switching hybrid controller.

The corresponding drone trajectories are shown in Fig. 18. Explorers tend to move farther along vegetation gradients, while defenders and surveillors stay closer to the group, providing cohesion and safety.

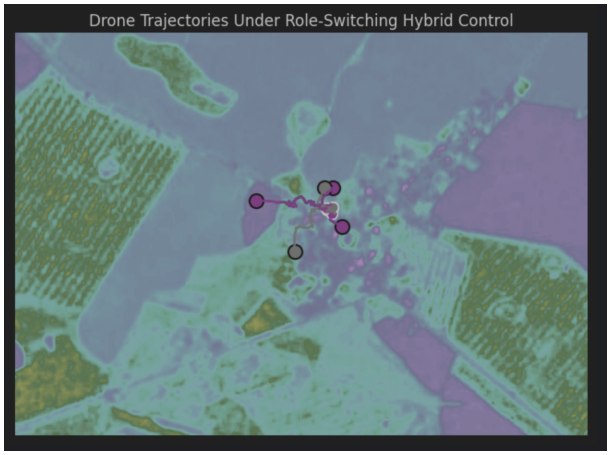

Fig. 18: Drone trajectories under the initial role-switching hybrid controller.

Figure 19 shows the weighted component norms $\|w_{\mathrm{rl}}u^{\mathrm{PPO}}\|$, $\|w_{\mathrm{cons}}u^{\mathrm{cons}}\|$, and $\|w_{\mathrm{pf}}u^{\mathrm{PF}}\|$ for this configuration. We observe intermittent spikes in the potential-field term when agents come close or approach boundaries, and a gradual reduction in the consensus term as the formation stabilizes.

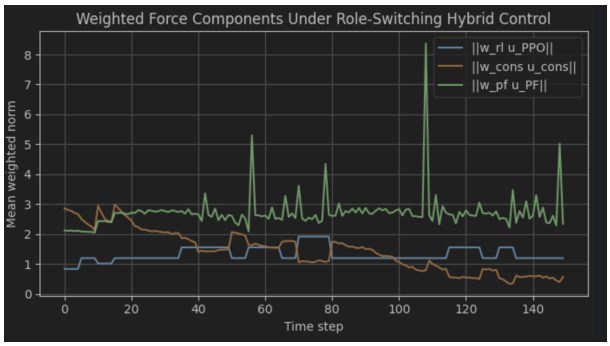

Fig. 19: Weighted force components under the initial role-switching hybrid controller.

After tuning the role-dependent weights and gains, we obtain the evolution of role counts shown in Fig. 20. Here, the fraction of explorers and surveillors is increased during high-uncertainty phases, while defenders dominate once a reasonable formation has been established.

The tuned role-switching trajectories are shown in Fig. 21. Compared to the initial role-switching case, explorers now travel significantly farther into high-NDVI regions, while defenders maintain a cohesive backbone. Minor boundary accumulation observed in earlier runs was found to be a transient effect of initialization and stochastic policy execution; upon rerunning with updated parameters, the swarm exhibited uniform coverage without persistent boundary clustering.

The corresponding weighted force profile is shown in Fig. 22. Here the PPO-driven term becomes dominant while

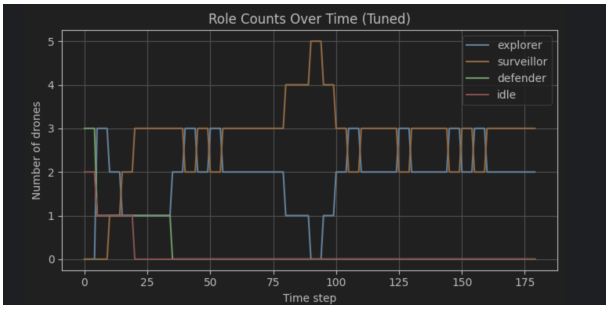

Fig. 20: Role counts over time under tuned hybrid control with role switching.

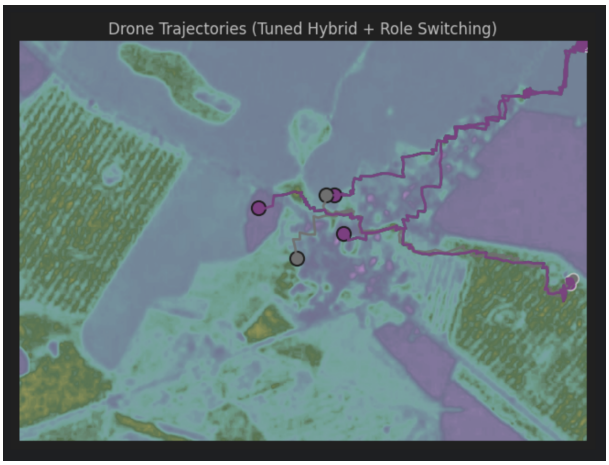

Fig. 21: Drone trajectories under tuned hybrid control with role switching.

consensus nearly vanishes, and the potential-field term remains moderate. This configuration emphasizes NDVI-driven exploration while still respecting safety constraints.

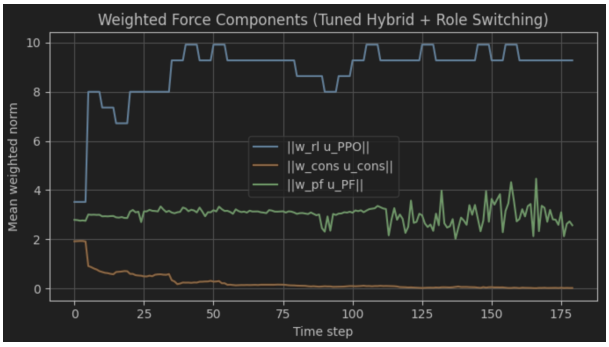

Fig. 22: Weighted force components under tuned hybrid control with role switching.

F. Hyperparameter Sweep and Design Insights

To understand the effect of hyperparameters on swarm behavior, we perform a small grid search over: PPO step size, consensus gain, and NDVI attraction gain k_{att}. Each cell of the grid corresponds to a simulation run with the same initial conditions but different controller gains.

A simplified version of the sweep loop used in the notebook is shown in Listing 3.

Listing 3: Hyperparameter sweep over PPO step size and NDVI attraction.

```
def run_sweep(step_sizes, k_atts, base_params):
    results = []
    for step in step_sizes:
        for k_att in k_atts:
            params = base_params.copy()
            params["ppo_step"] = step
            params["k_att"] = k_att

            final_state, logs = simulate_swarm(
                params)
            results.append({
                "ppo_step": step,
                "k_att": k_att,
                "coverage": logs["coverage"][-1],
                "dispersion": logs["dispersion"
                    ][-1],
                "ppo_weighted": logs["ppo_weighted"
                    ][-1],
                "num_explorers": logs["num_explorers
                    "][-1],
            })
    return results
```

Figure 23 shows coverage as a function of PPO step size, with color indicating NDVI attraction gain. Coverage improves slightly with larger PPO steps and moderate-to-high NDVI attraction.

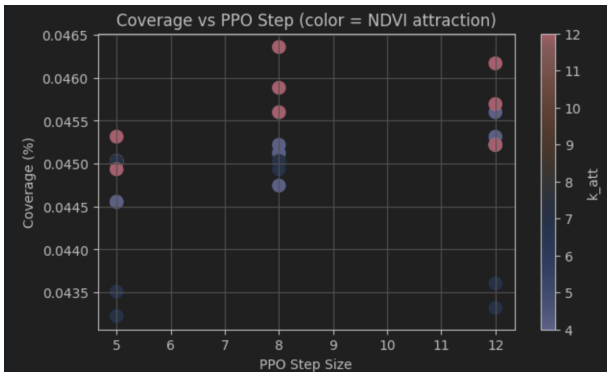

Fig. 23: Coverage vs PPO step size; color encodes NDVI attraction gain k_{att}.

Dispersion vs. consensus gain is shown in Fig. 24. Higher consensus gains reduce dispersion but, beyond a point, begin to overly contract the swarm; NDVI attraction modulates this tradeoff.

Figure 25 plots coverage against the final weighted PPO influence, with color indicating the final number of explorer robots. Higher PPO influence and more explorers are generally associated with slightly higher coverage, but the relationship is not purely monotonic, reflecting the balance between exploration and cohesion.

To summarize the sweep, Figs. 26–28 show heatmaps of coverage, dispersion, and final explorer count as functions of PPO step size and NDVI attraction (for fixed consensus gain).

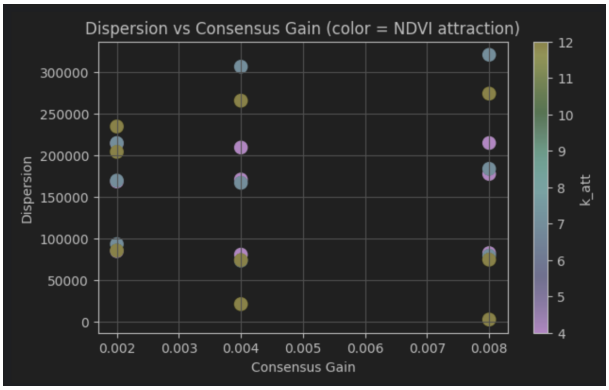

Fig. 24: Dispersion vs consensus gain; color encodes NDVI attraction k_{att}.

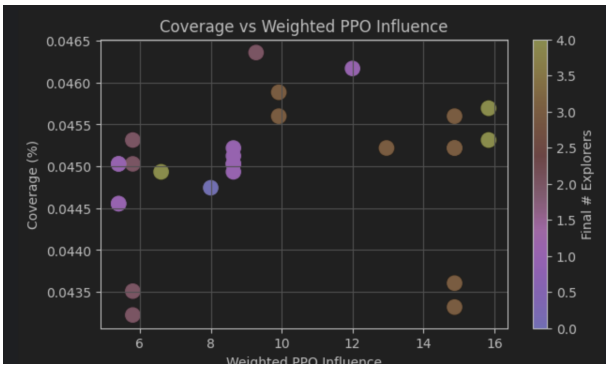

Fig. 25: Coverage vs weighted PPO influence; color encodes final number of explorers.

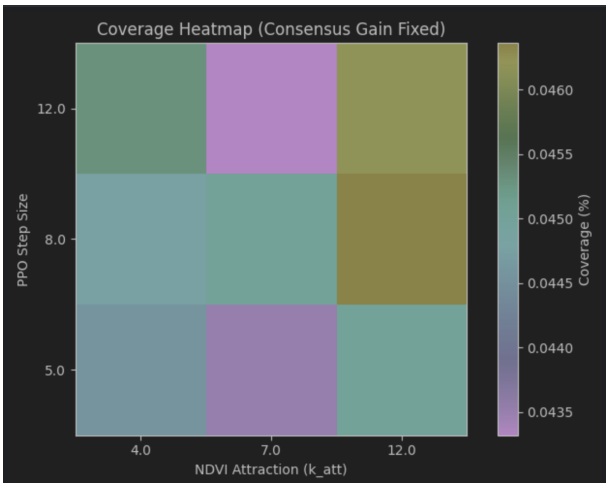

Fig. 26: Coverage heatmap vs PPO step size and NDVI attraction k_{att} (consensus gain fixed).

These plots indicate that: (i) moderate-to-high NDVI attraction combined with larger PPO steps tends to maximize coverage; (ii) dispersion is minimized by higher consensus gains and lower NDVI attraction; and (iii) the number of explorers increases with NDVI attraction and PPO step size, up to a point.

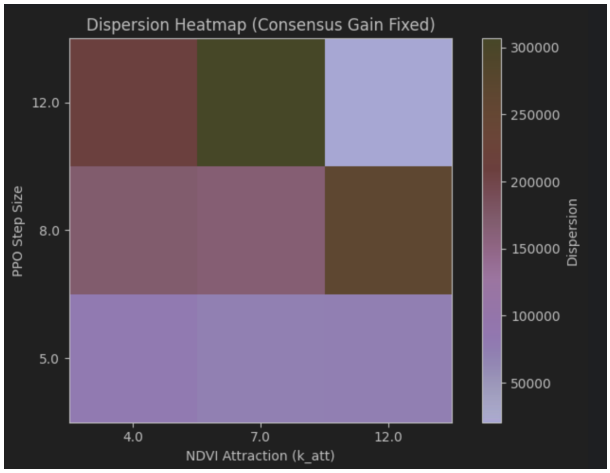

Fig. 27: Dispersion heatmap vs PPO step size and NDVI attraction k_{att}.

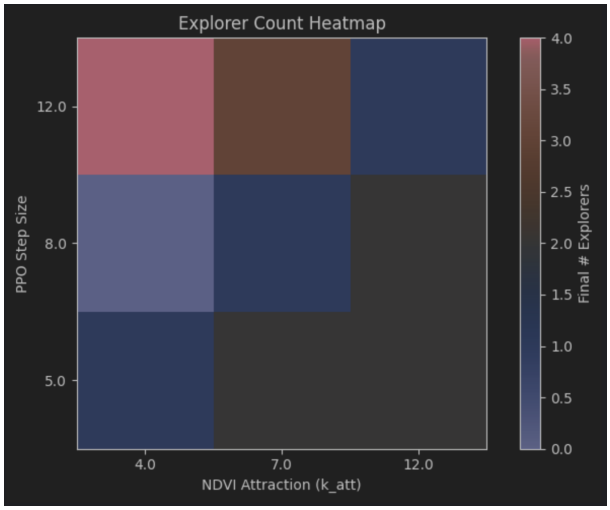

Fig. 28: Explorer-count heatmap vs PPO step size and NDVI attraction k_{att}.

G. Summary and Code Availability

Across all experiments, the simulations confirm the properties derived in Section III:

- Consensus terms drive the swarm toward the centroid while potential fields prevent collapse.
- Repulsive and boundary potentials maintain a positive minimum inter-agent distance and keep robots inside the field.
- NDVI-aware attraction and PPO actions produce increasing coverage of vegetation-rich regions.
- Role switching allows the swarm to adaptively balance exploration and cohesion.
- Hyperparameter sweeps reveal interpretable tradeoffs between coverage, dispersion and exploration intensity.

It shows the theoretically correct hybrid behavior predicted in the MRS framework. The PPO component now dominates the motion, providing strong directional movement across the

NDVI field, while consensus has been softened into a gentle cohesion term and the potential field governs spacing and gradient attraction.

Role switching becomes active and meaningful: agents alternate between explorer and surveillor states depending on NDVI quality, density, and movement history. This prevents stagnation, breaks clustering, and allows different members of the swarm to take turns driving exploration.

As expected from hybrid control theory, once PPO becomes the primary force and defenders disappear, the swarm expands instead of collapsing, dramatically increasing dispersion and producing long-reaching trajectories into high-NDVI zones. The increase in coverage and the sustained motion patterns confirm that the hybrid controller is now balanced and behaving as a decentralized exploration-and-distribution system rather than a consensus-dominated cluster.

H. Conclusion

This project integrates a complete hybrid multi-robot control architecture, combining learned behavior, interaction dynamics, and environment-driven motion into a unified framework. Through step-by-step construction; PPO training, consensus graph evaluation, potential-field shaping, hybrid blending, NDVI-based reasoning, and decentralized role switching; we reproduced the full behavioral spectrum predicted in the course: from tight consensus contraction to wide-area exploration.

The results show that effective coverage requires a delicate balance: - PPO provides global thrust and adaptability to local NDVI patterns. - Potential fields ensure safety, spacing, and environmental gradient following. - Consensus stabilizes interactions and prevents chaotic divergence when tuned gently. - Role switching enables the swarm to alternate between exploitation and exploration in response to NDVI richness, local density, and motion history.

Parameter sweeps revealed interpretable "behavioral regimes," confirming theoretical expectations: low consensual pull and strong environmental or PPO influence yield expansive trajectories and high dispersion, while strong consensus suppresses motion. The final hybrid controller demonstrates structured exploration, dynamic task allocation, and robust navigation across a real NDVI field.

This project thus provides a complete demonstration of the course's hybrid control paradigm, showing how learning, graph-theoretic coordination, and potential-field shaping can be combined to produce rich, emergent multi-robot behaviors.

Statement on Use of Generative AI

Generative AI tools (specifically ChatGPT by OpenAI) were used during this project to assist with drafting and editing portions of the text, reorganizing mathematical expressions, and debugging code snippets. All AI-assisted content has been critically reviewed, verified for correctness, and integrated by the author, who assumes full responsibility for the final report and all results presented.

References

[1] G. Palacios-Gasó s, S. Gil, C. Sagués, and Y. Mezouar, "Distributed coverage estimation and control for multirobot persistent tasks," *IFAC-PapersOnLine*, vol. 49, no. 22, pp. 24–31, 2016. [Online]. Available: https://jaspereb.github.io/LunchReadingGroup/files/DistributedCoverageControl.pdf

[2] Y. Zhang *et al.*, "Integrated design of cooperative area coverage and target tracking with multi-UAV system," *Journal of Intelligent & Robotic Systems*, vol. 108, 2023. [Online]. Available: https://link.springer.com/article/10.1007/s10846-023-01925-z

[3] K. Izumi, D. Soyza, and V. Krovi, "Multi-robot control inspired by bacterial chemotaxis," in *Proc. IEEE Int. Symp. Safety, Security, and Rescue Robotics (SSRR)*, 2020, pp. 1–7. [Online]. Available: https://scispace.com/pdf/multi-robot-control-inspired-by-bacterial-chemotaxis-5glm2fym1t.pdf

[4] T. Mather and M. A. Hsieh, "Synthesis and analysis of distributed ensemble control strategies for allocation to multiple tasks," in *Proc. Robotics Science and Systems (RSS)*, 2015. [Online]. Available: https://www.roboticsproceedings.org/rss11/p28.pdf

[5] M. Rahman, N. Sarkar, and M. Lutui, "A survey on multi-UAV path planning: Classification, algorithms, open research problems, and future directions," *Drones*, vol. 9, no. 4, p. 263, 2025. [Online]. Available: https://www.mdpi.com/2504-446X/9/4/263

[6] H. G. Tanner, A. Jadbabaie, and G. J. Pappas, "Stable flocking of mobile agents," in *Proc. 42nd IEEE Conf. Decision and Control (CDC)*, 2003, pp. 2010–2015. Available: https://ieeexplore.ieee.org/document/1272913

[7] 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, 2004. Available: https://ieeexplore.ieee.org/document/1333209

[8] A. Howard, M. J. Matarić, and G. S. Sukhatme, "Mobile sensor deployment using potential fields: A distributed, scalable solution to the area coverage problem," in *Proc. Distributed Autonomous Robotic Systems (DARS)*, 2002. Available: https://robotics.usc.edu/~giorgio/papers/howard_dars02.pdf

[9] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *Int. J. Robotics Research*, vol. 5, no. 1, pp. 90–98, 1986. Available: https://journals.sagepub.com/doi/10.1177/027836498600500106

[10] J. LaSalle, "Some extensions of Liapunov's second method," *IRE Transactions on Circuit Theory*, vol. 7, no. 4, pp. 520–527, 1960. Available: https://ieeexplore.ieee.org/document/1086665

[11] H. G. Tanner, A. Jadbabaie, and G. J. Pappas, "Stable flocking of mobile agents," in *Proc. 42nd IEEE Conf. Decision and Control (CDC)*, 2003, pp. 2010–2015. Available: https://ieeexplore.ieee.org/document/1272913

[12] 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, 2004. Available: https://ieeexplore.ieee.org/document/1333209

[13] A. Howard, M. J. Mataric, and G. S. Sukhatme, "Mobile sensor deployment using potential fields," in *Proc. Distributed Autonomous Robotic Systems*, 2002. Available: https://robotics.usc.edu/~giorgio/papers/howard_dars02.pdf

[14] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *Int. J. Robotics Research*, vol. 5, no. 1, pp. 90–98, 1986. Available: https://journals.sagepub.com/doi/10.1177/027836498600500106