Large-Scale Agents Cooperative Control in High-Dimensional Environments Based on Deep Reinforcement Learning

Olga Krivorotko, Ilya Markov, Anastasiia Murzina, Konstantin Pronin, Yakov Scherbina

Sirius University · MEPhI · ITMO University

International Conference on Cyber Warfare, Security & Space Computing

St. Petersburg, 26–28 October 2026

PROBLEM SETTING AND SCOPE

Trend.
Defense, surveillance, and logistics are moving from single-agent operations to large-scale UAV/robotic swarms that execute cooperative missions in high-dimensional environments.

MARL promise.
Multi-Agent Reinforcement Learning produces emergent cooperative behavior — encirclement, flanking maneuvers, coordinated actions — without explicitly programming every scenario.

But: MARL typically assumes the environment is already given in a structured form (waypoints, low-dimensional state vectors).

Challenges

1.Perception quality limits scaling
Policy, reward, and safety constraints all depend on how reliably the swarm perceives the environment.

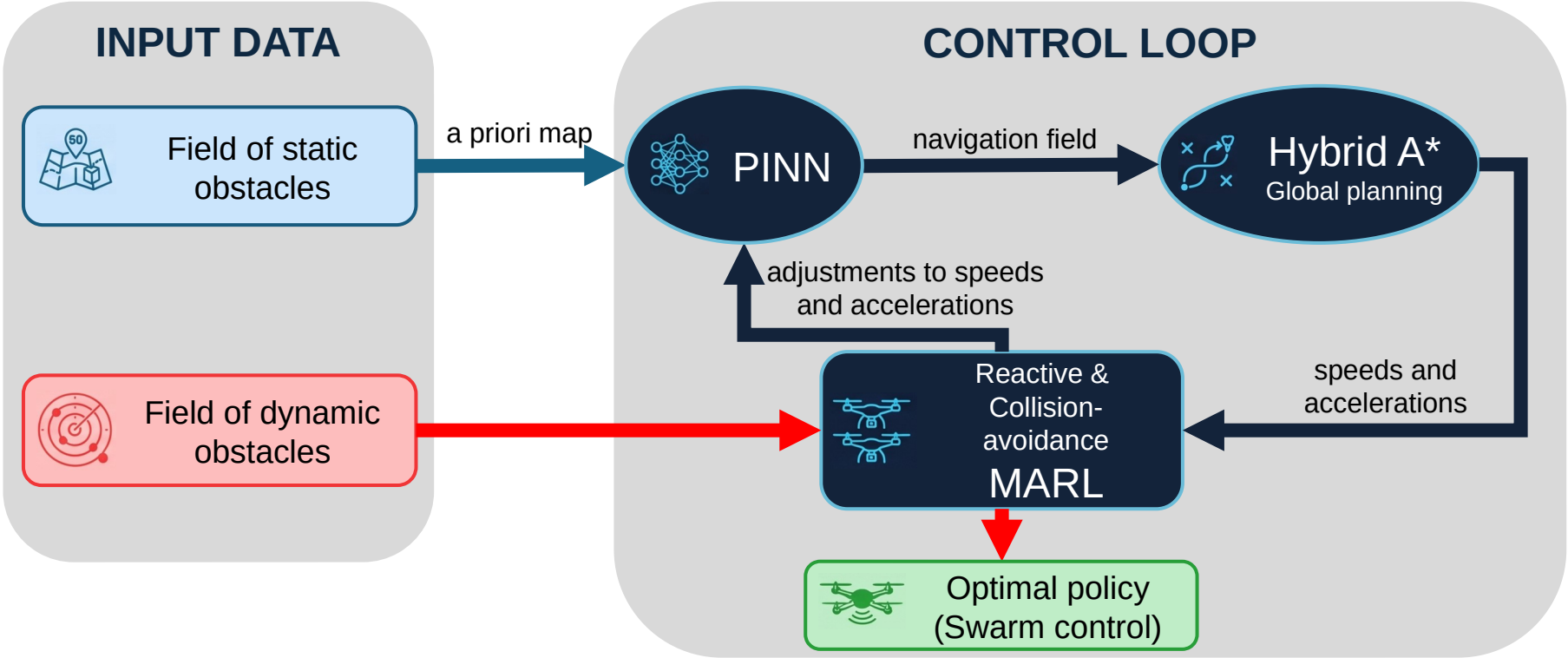
2.High-dimensional state space
Agents must operate in large, complex state spaces and still produce real-time control policies.

3.Individual vs. collective objectives
Balancing each agent's local contribution with the mission-level collective goal

Scope of this work

- Obstacles are provided as geometric primitives by an upstream detection/segmentation pipeline.
- We solve the local planning problem: traversing the segment between two consecutive waypoints when an obstacle blocks the direct path.

HIERARCHICAL CONTROL ARCHITECTURE



PINN (3D): compute time ~5.7 ms on GPU, field error 1.15%

Hybrid A* (2D): 30/30 randomized maps

MARL: attention-based decentralized control

MOTION MODEL AND HFC MODES

ITMO

LEADER–FOLLOWER SCHEME

Leader $i = 0$ follows the A* route on an occupancy map; followers $i = 1, ..., N$ keep formation offsets δ_i and adapt to local constraints. Planar workspace $\Omega \subset \mathbb{R}^2$ with static obstacles.

Second-order dynamics with filtering [3]

$$\dot{p}_i = v_i, \quad \dot{v}_i = u_i, \quad \dot{u}_i = (a_i - u_i) / \tau_i$$

p_i, v_i: position, velocity

a_i: control acceleration; u_i: filtered acceleration

$\tau_i > 0$: time constant

HYBRID FORMATION-CHAIN (HFC) MODES

Mode	Reference point
F Formation	formation target
C Chain	point on the leader's trace
R Recovery	local recovery waypoint
G Go-to-Formation	intermediate go-to-formation point

SWITCHING LOGIC

Transition	Condition
F → C	formation target obstructed or unsafe
C → R	chain point unreachable or agent stuck
C → G	chain point reachable, region free
R → C	leader's trace reached or recovery done
G → C	formation target unsafe again
G → F	target reachable and agent close to it

Leader–follower: formation offsets and consensus [3], [9]

HFC: mode switching introduced in this work

Effect: formation relaxed only where needed

FOLLOWER CONTROL LAW

$$a_i = F_i^{form} + F_i^{nav} + F_i^{obs} + F_i^{drone}$$

Formation (consensus)

$F_i^{form} = w_{form}(m_i)\ k_{form}\ \Sigma_j\ g_{ij}\ (p_j^{ref} - p_i^{ref})$

$w_{form} = 1$ in mode F, 0 otherwise

Navigation

$F_i^{nav} = k_{nav}\ (p_i^{ref} - p_i) + k_{damp}\ (v_i^{ref} - v_i)$

v_i^{ref} by finite differencing of p_i^{ref}

Obstacle repulsion (APF)

$F_i^{obs} = \Sigma_k\ 1\{d_k < r_{infl}\}\ k_{obs}\ (1/d_k^{eff} - 1/r_{infl})\ n_k\ /\ (d_k^{eff})^2$

$d_k^{eff} = \max\{d_k, r_{reg}\}$, $r_{reg} = 0.05$ m prevents singularities

Inter-agent repulsion

Same form over neighbours with $r_{ij} = \|p_i - p_j\|$ and k_{drone}

Active in all modes; prevents collisions

[3] Chang et al., Front. Neurosci. 18, 1367248 (2024). [9] Ma et al., Drones 9(6), 390 (2025).

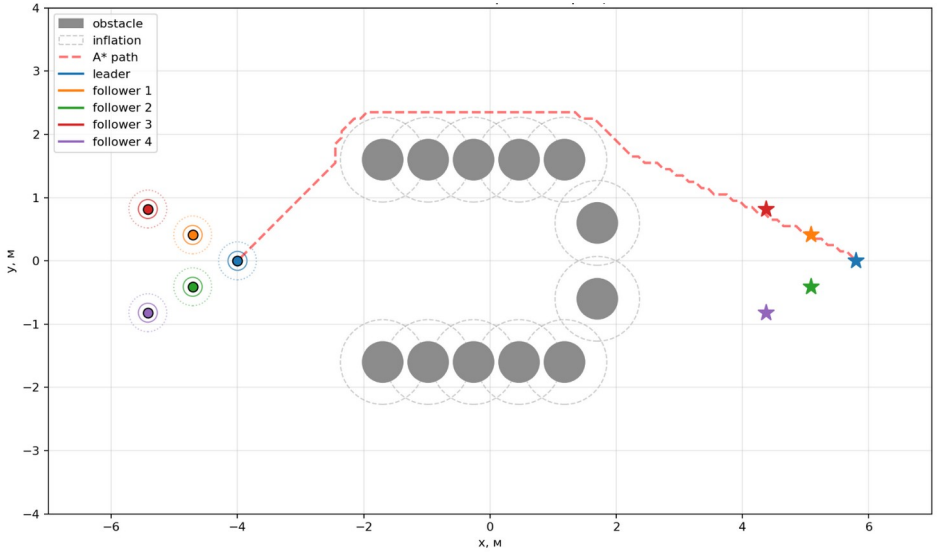
Mode-dependent: formation pull only in mode F

Always on: obstacle and inter-agent repulsion

HFC-A*: A* route + offsets + consensus/APF + switching

NUMERICAL EXPERIMENTS (2D)

U-SHAPED LOCAL-MINIMUM TRAP



$\min d_{agents} = 0.38$ m > $\varepsilon_{agents} = 0.30$ m

min obstacle clearance 0.35 m > $\varepsilon_{obs} = 0.15$ m

30 RANDOMIZED MAPS

5 agents; three clusters of four cylinders (r = 0.25 m); identical seeds for all methods

Method	Success	Time, s
HFC-A*–APF	**30/30**	**15.57 ± 1.34**
A*–APF (no HFC)	28/30	16.12 ± 1.66
Traditional APF	29/30	16.74 ± 2.69

Success: all agents reach their final slots at rest without violating the safety thresholds.

Time: swarm completion, mean ± s.d. over successful runs.

Full method: 30/30, fastest and least variable

Ablation: fails twice: one deadlock, one collision

Classic APF: largest time spread from local-minimum escapes

DECENTRALIZED MULTI-AGENT RL METHOD

FRAMEWORK & TRAINING

Problem Setting

Dec-POMDP & CTDE
(Centralized training, Decentralized execution)
Each drone learns decentralized navigation & collision-avoidance policies

Policy optimization

PPO with clipped objective; joint reward: progress, safety, coordination

Domain Randomization

Varying mass/inertia, sensor noise & dynamic configurations to bridge Sim-to-Real gap

AGENT OBSERVATION (o_i^t)

Self-State & Goal (q_i, e_{goal})
3D-position, velocity, orientation, body rates + goal vector

Neighbors (E_{neigh}):
relative state ($\Delta p_{ij}, \Delta v_{ij}, d_{ij}$) of K nearest drones

Obstacles (E_{obs}):
relative state ($\Delta p_{ik}, r_k, v_k$) of M nearest obstacles

COMPUTATIONAL PIPELINE

Raw Obs → MLP Encoders → Actor Network

Scaled Dot-product attention

Query (Q_i): self-state,
Keys/Values (K_j, V_j): neighbors & obstacles

$$\alpha_{ij} = \frac{\exp(Q_i K_j^T/\sqrt{d})}{\sum \exp(Q_i K_m^T/\sqrt{d})}$$

(relevance weights)

Context vector:

$$c_i = \sum \alpha_{ij} V_j$$

Combined state:

$$h_i = [q_i, c_i]$$

GAUSSIAN POLICY
$\pi_\theta(a_i|h_i)$

Outputs a continuous 4D control vector

$$a_i = [T, \omega_x, \omega_y, \omega_z]^T$$

T: thrust,
$\omega_x, \omega_y, \omega_z$: angular velocities

Decentralized execution: policies use local observations only

Attention mechanism: Scalable with varying swarm density; dynamically prioritizes critical hazards

Low-level Control: Gaussian 4D control vector links directly to low-level onboard controllers (PID)

PINN: EIKONAL PLANNING FIELD

ITMO

PROBLEM SETTING

Reach–avoid for a swarm

N agents, each a 12-DoF rigid body; minimum separation d_{safe}; common goal.

Requirement

One solution for a whole family of obstacle configurations: an unseen map needs no re-optimization.

Decomposition

(i) planning field whose anti-gradient gives the direction of motion;

(ii) differentiable controller that tracks it under full dynamics.

EIKONAL PLANNING FIELD

Cost-to-go field V

$$|\nabla V(x)| = 1 \,/\, c(x), \quad V(x_g) = 0$$

c = 1 in free space, c = ε ≪ 1 inside obstacles

$-\nabla V \,/\, |\nabla V|$ yields collision-free geodesics around obstacles

Reference fields: fast marching, re-solved for every map

The operator learns the whole map O → V instead

FOURIER NEURAL OPERATOR

Inputs (48³ grid)
coordinates, occupancy mask, distance-to-goal

FNO 4 Tucker-factorized layers, 16 modes
$\hat{V} = G_\theta(O)$

Loss with gradient matching

$$L = \|\hat{V} - V\|^2 + \lambda \|\nabla\hat{V} - \nabla V\|^2_{\Omega_{free}}$$

λ = 0.5; supervised fit to fast-marching references, not Eikonal-residual training

PINN: physics-informed neural operator; a new map costs one forward pass

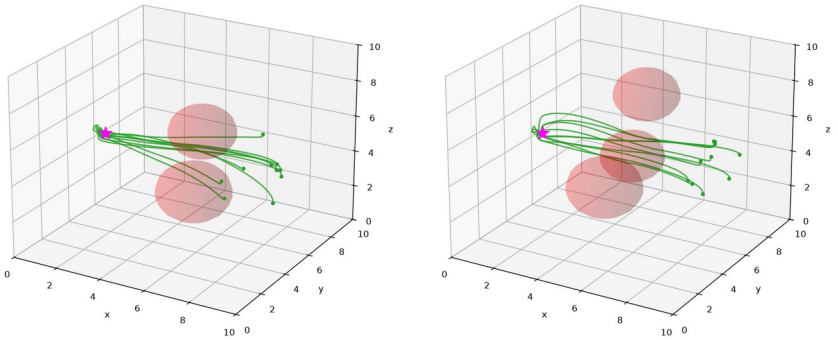
Gradient matching: the gradient is the steering signal

Data: 256 training maps, 2–4 spheres

DIFFERENTIABLE SWARM CONTROLLER

ITMO

SINGLE AGENT, CLOSED LOOP



Single-agent 12-DoF flights guided by the planning field around spherical obstacles; ★ marks the goal.

State: position, velocity, attitude, body rates. **Control**: collective thrust and torques.

POLICY

Position-invariant attention policy

$u_i = \pi_\theta(v, \eta, \omega, d(p), \mathrm{Attn}_K(\Delta p, \Delta v))$

$d(p) = -\nabla\hat{V} \,/\, |\nabla\hat{V}|$ read from the operator field by differentiable trilinear interpolation; K = 6 nearest neighbours

TRAINING

Differentiable rollout

$x_{n+1} = x_n + \Delta t\, f(x_n, \pi_\theta)$; $\nabla_\theta J$ by backpropagation through the integrator: no expert teacher, no model-free RL

Trajectory cost

Field tracking (tapered near the goal), separation, obstacle avoidance, effort, terminal hover

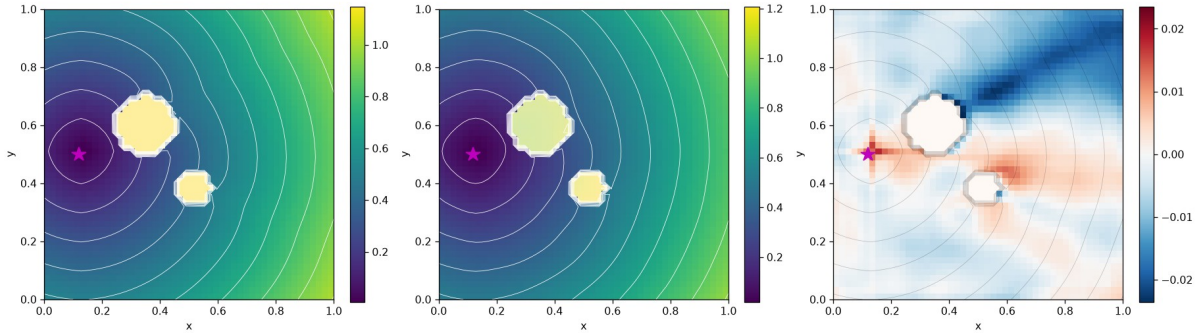
Exact gradients: low-variance, end-to-end through the dynamics

Goal region: a point goal collapses the swarm

Shared field: one planning field for all agents

PINN ACCURACY ON UNSEEN MAPS

GOAL-PLANE SLICE, UNSEEN CONFIGURATION



Fast marching (reference) *Neural operator* *Signed error $\hat{V} - V$*

Metric (32 unseen maps)	Supervised L2	+ gradient matching
Field relative L2 error	2.65%	1.15%
Median gradient angle error	13.33°	1.75°
Kinematic goal reaching	89.5%	99.8%

SPEED PER NEW MAP

One forward pass

GPU: 5.7 ms (RTX 5060 Ti)

CPU: 92.6 ms vs 65.7 ms for fast marching

The advantage comes from GPU parallelism and generalization across maps, not from a cheaper single solve.

Setup

256 training maps, 32 held-out; 48^3 grid; 2–4 spheres per map

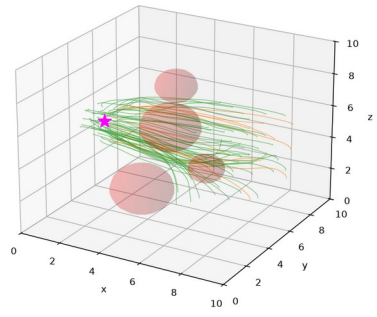
Error structure: smooth far-field offset, sharp residuals at obstacle boundaries

Gradient term: angle error 13.33° → 1.75°

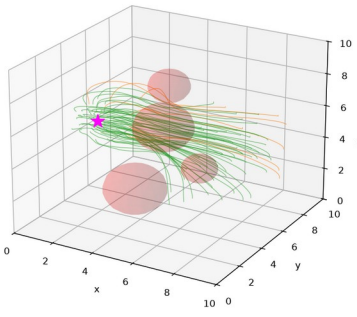
Generalization: 32 out-of-distribution maps

CLOSED-LOOP SWARM: 64 AGENTS, 12-DoF

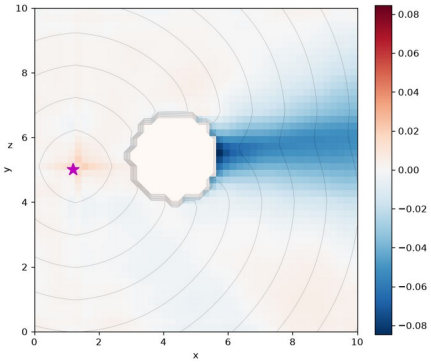
SWARM OF 64 AGENTS, ONE UNSEEN ENVIRONMENT



Fast-marching field



Neural-operator field



Signed error $\hat{V} - V$

Field	N	Reach	Obstacle	Inter-agent
Fast marching (upper bound)	64	96%	3%	11%
Neural operator	**64**	**95%**	**16%**	**7%**

EVALUATION

Paired design

48 unseen environments, identical initial states for both fields; single agent: 98% vs 94% reach

Robustness of the estimate

5 resampled sets: reach 95.9 ± 0.9% (operator) vs 95.3 ± 1.2% (reference)

Collisions

Inter-agent contacts are transient; the final configuration is collision-free

Near-lossless: reach within ~1 pp of the exact field

Goal region: prevents swarm collapse

Open gap: obstacle hits 16% vs 3%: gradient error at boundaries

CONCLUSIONS

COORDINATION

30/30

Hybrid A*–APF guides formations through constrained passages; safety margins kept on all randomized maps

SCALABILITY

64 agents

Attention over nearest neighbours and obstacles; one PINN field shared by the swarm under 12-DoF dynamics

RESILIENCE

≈1 pp

Decentralized execution; PINN replaces the numerical solver in goal reaching with less dependency on remote infrastructure

Validation: the layers were evaluated separately (Hybrid A*–APF in 2D, PINN with the swarm controller in 3D); end-to-end integration of all three layers is the next step.

Next: dynamic obstacles, heterogeneous swarms

Perception: integrate perception uncertainty

Deployment: bridge simulation and real flight

Thank you for your attention!

ITMO more than a UNIVERSITY