

# Modeling and Predicting LTE-Induced Learnability Cliffs in RL-Controlled UAV Swarms

Stephen Otero, Dwight Makaroff, Derek Eager

Department of Computer Science

University of Saskatchewan, Saskatoon, Canada

yat251@usask.ca, makaroff@cs.usask.ca, eager@cs.usask.ca

**Abstract:** Centralized reinforcement learning (RL) over Long-Term Evolution (LTE)/4G networks faces a fundamental challenge: uplink losses thin the reward signal (the per-step feedback that guides training) and stall learning even when most packets arrive. This paper quantifies this effect for rural search-and-rescue (SAR) swarms via Centralized Proximal Policy Optimization (CPPO).

A four-component rural LTE channel model (path loss, shadowing, block error rate (BLER), and zero-order-hold (ZOH) replay) is validated in the ns-3 LENA module and coupled to a Godot 4 drone simulator. Sixty-six scenarios (2 ideal baselines, 4 LTE tiers  $\times$  6 profiles  $\times$   $N \in \{3, 6\}$ , and 16 time-of-day configurations) are each run twice on the same channel: once with CPPO, once with a lawnmower sweep. Two failure regimes emerge: a *radio hard limit* (harsh/extreme LTE) where neither controller confirms any victim; and an *RL learnability cliff* (mild/medium LTE) where CPPO falls 4.5–12% in coverage below the lawnmower (a learning failure, not a radio one) despite identical conditions. Plotting discovery against SINR per tier exposes both boundaries for this scenario suite: a learnability cliff near 14.7–23.0 dB and a hard limit below 7.1 dB.

**Index Terms:** LTE, reinforcement learning, UAV swarm

## I. INTRODUCTION

In this paper, we consider the problem of using reinforcement learning (RL) for control of a drone swarm in a context of potentially unreliable LTE communication. We focus on search-and-rescue (SAR) as a specific application domain. A single centralized RL controller (CPPO) coordinates the swarm over a 4G LTE link, sending navigation commands to drones and receiving their sensor observations. When the channel degrades, lost uplinks withhold observations, lost downlinks render commands stale, and uplink packet loss directly thins the reward signal that guides training, stalling learning even when coverage appears normal. Standard link-quality tests [1] evaluate signal-to-interference-plus-noise ratio (SINR), BLER, delay, and data rate separately, but RL training depends on a single reward shaped by all four jointly: a link that passes every individual check can still starve the training signal.

A four-component rural LTE channel model is built in ns-3 LENA, validated against published 3GPP rural air-to-ground data [2], [3], and connected to a Godot 4 physics engine. The 66-scenario matrix is run twice on the same channel, once with CPPO and once with a non-learning lawnmower baseline. This separates two failure modes: a radio hard limit where both controllers fail equally, and an RL learnability cliff where CPPO falls 4.5–12% in coverage below the lawnmower even when most packets arrive.

## Contributions.

- 1) *Validated LTE testbed:* a four-component rural LTE model (path loss, shadowing, BLER, ZOH) in ns-3 LENA coupled to Godot 4, reporting joint network statistics and RL outcomes (Sec. III–IV).
- 2) *Learnability cliff via the lawnmower baseline:* a non-learning lawnmower on the same channel and testbed as the RL agent separates radio failure from learning failure, exposing a regime where centralized RL falls below a simple sweep over impaired LTE (Sec. VI-B).
- 3) *Experimental analysis of RL learnability cliffs under LTE impairment* (Sec. VI-G): plotting victim discovery against per-tier average SINR reveals two failure boundaries for this scenario suite: above 23.0 dB, the link remains learnable; below 7.1 dB, the radio hard limit prevents discovery by any controller; the learnability cliff lies between these thresholds.

## II. RELATED WORK

**Cooperative multi-agent reinforcement learning (MARL).** Standard cooperative methods (QMIX [4], Multi-Agent DDPG (MADDPG) [5], Multi-Agent Proximal Policy Optimization (MAPPO) [6], Multi-Agent Transformer (MAT) [7]) assume centralized training with decentralized execution (CTDE) over a lossless link; this work uses CPPO with centralized execution (CTCE), where all drone commands travel over the LTE link. PPO stability studies [8], [9] confirm sparse rewards cause failures, but none link sparsity to radio impairments.

**Learned communication.** CommNet [10], DIAL [11], IC3Net [12], TarMAC [13], and I-MAC [14] learn selective message sharing, but assume a reliable physical link. Radio-layer packet loss withholds training feedback before any learned protocol can compensate: this is the learnability cliff.

**Multi-UAV RL.** Unmanned aerial vehicle (UAV) coordination via deep RL [15], [16] assumes either an ideal lossless link or a simple path-loss model, rarely measuring SINR or BLER; air-to-ground surveys [17] characterize propagation but do not connect it to RL training outcomes.

**Network-aware RL.** Channel-aware reward design [3] and UAV network characterization [18] exist but neither links per-tier statistics to RL outcomes. Our previous work [19] shows 6% packet loss cuts single-drone PPO success by 66%, but used an artificial channel, one drone, and no non-RL baseline.

### III. LTE CHANNEL MODEL

A rural 4G LTE network serving an  $N$ -drone swarm on a dedicated partition is modeled. Two base stations (evolved NodeBs, eNBs; 2.5 km apart, 900 MHz Band 20, 10 MHz / 50 resource blocks, 46 dBm eNB / 23 dBm drone transmit power, equal resource sharing) cover a 5 km  $\times$  5 km area, a typical rural SAR footprint [20]. Propagation follows a rural macro-cell model with distance-dependent path loss, log-normal shadowing, and small-scale fading. Uplink and downlink traffic share the available LTE resources, creating contention as swarm size increases and channel quality varies across the terrain. Five channel tiers (one ideal reference plus four rural severities, independent of drone altitude) each set a path-loss exponent  $n$ , log-normal shadowing standard deviation  $\sigma_{\text{sh}}$  (dB), and per-tier packet-loss, delay, and bandwidth limits (Table I). The ideal tier is a clean reference; the four rural tiers (mild/medium/harsh/extreme) span the range of published real-world LTE drone measurements [2], [3].

TABLE I  
LTE CHANNEL TIERS AND THEIR PER-TIER RADIO PARAMETERS.

| Tier    | Loss (%) | RTT (ms) | BW (Mbps) | $n$ | $\sigma_{\text{sh}}$ (dB) | SINR (dB) | BLER (%) |
|---------|----------|----------|-----------|-----|---------------------------|-----------|----------|
| ideal   | 0        | 0        | 100       | 2.0 | 0                         | 34.9      | 0.10     |
| mild    | 3        | 64       | 15        | 2.4 | 3                         | 23.0      | 3.96     |
| medium  | 7        | 94       | 10        | 2.9 | 5                         | 14.7      | 10.7     |
| harsh   | 15       | 134      | 5         | 3.4 | 8                         | 7.1       | 22.9     |
| extreme | 25       | 184      | 3         | 4.0 | 12                        | -6.6      | 43.7     |

#### A. ns-3 LENA Simulator Validation

The model runs in ns-3.40 with the LENA module [21], implementing full LTE protocol layers (physical, MAC, and transport), proportional-fair scheduling, and automatic retransmission. Thirty 60-second calibration runs per tier (3 client nodes per cell) are compared against rural air-to-ground measurements [3] and published UAV channel models [2] (Table II). The simulator matches SINR, BLER, and RTT to within 1.6 dB, 1.7%, and 9 ms, confirming it as a reliable stand-in for rural LTE.

TABLE II  
NS-3 LENA VALIDATION AGAINST PUBLISHED RURAL LTE FIELD MEASUREMENTS ([3], [2]). SIM: NS-3 PER-TIER MEAN; MEASURED: FIELD MEAN;  $\Delta$ : ABSOLUTE DIFFERENCE.

| Tier    | SINR (dB) |          |          | BLER (%) |          |          | RTT (ms) |          |          |
|---------|-----------|----------|----------|----------|----------|----------|----------|----------|----------|
|         | Sim       | measured | $\Delta$ | Sim      | measured | $\Delta$ | Sim      | measured | $\Delta$ |
| ideal   | 34.9      | 33.8     | 1.1      | 0.10     | 0.20     | 0.1      | 0        | 0        | 0        |
| mild    | 23.0      | 22.1     | 0.9      | 3.96     | 4.30     | 0.3      | 64       | 71       | 7        |
| medium  | 14.7      | 13.4     | 1.3      | 10.7     | 9.0      | 1.7      | 94       | 102      | 8        |
| harsh   | 7.1       | 5.5      | 1.6      | 22.9     | 21.3     | 1.6      | 134      | 143      | 9        |
| extreme | -6.6      | -5.9     | 0.7      | 43.7     | 42.0     | 1.7      | 184      | 178      | 6        |

### IV. SIMULATION PLATFORM AND SAR SCENARIO

#### A. End-to-End Architecture

Three components run in synchronized 14-second steps: the ns-3 LENA radio model, the Godot 4 physics simulator, and the CPPO controller. The CPPO controller is the sole decision maker: each drone carries only an onboard sensor suite, a proportional-integral-derivative (PID) autopilot, and an LTE radio transceiver.

**Uplink: drone  $\rightarrow$  CPPO controller.** At each 14-second boundary, drone  $i$  packages its state into a UDP uplink datagram:

grid cell  $c_t^i \in \{0 \dots 624\}$ , altitude  $a_t^i \in \{0, 1, 2\}$  (40, 80, 120 m), binary detection flag  $d_t^i \in \{0, 1\}$  (output of the onboard optical or thermal camera; detection probability  $p_{\text{det}}(z)$ , Eq. (1)), battery fraction  $b_t^i$ , and LTE telemetry (SINR $_t^i$ , BLER $_t^i$ , RTT $_t^i$ ), transmitted via drone radio  $\rightarrow$  eNB  $\rightarrow$  evolved packet core (EPC)  $\rightarrow$  backhaul  $\rightarrow$  CPPO controller. If the uplink packet for drone  $i$  is lost (per-tier BLER, Table I), the CPPO controller substitutes that drone's most recent received observation; the joint command is then issued from whatever frames arrived and no decision step is deferred.

**CPPO controller.** The consolidated joint observation  $o_t$  (Sec. IV-C) is evaluated by the shared policy network; the resulting per-drone waypoints are packed into downlink datagrams and dispatched over the backhaul link.

**Downlink: CPPO controller  $\rightarrow$  drone.** Each per-drone command travels the downlink (backhaul  $\rightarrow$  EPC  $\rightarrow$  eNB  $\rightarrow$  drone radio). A lost downlink frame triggers ZOH at the drone, keeping the drone physically moving. The PID autopilot handles all low-level flight; the CPPO controller never issues velocity or attitude commands. Godot 4 runs drone physics at 100 Hz on a 5 km  $\times$  5 km terrain (200 m cells), with 60-minute episodes from a forward operating base (FOB).

#### B. SAR Mission, Sensing, and Belief

One victim is placed randomly in the terrain, amid random trails, rivers, and hills. Probability of Success (POS)=POA $\times$ POD [20], [22]: POA (Probability of Area) is the probability the victim lies within the searched region (covered grid fraction), and POD (Probability of Detection) is the per-overflight detection rate, scaled by altitude and terrain conditions. The CPPO controller begins with a bell-curve belief map centered on the victim's a priori position (the initial position estimate, uncertain even before any victim movement;  $\sigma_{\text{prior}}=750$  m, peak 0.10):  $B_0(c)=0.10 \exp(-d_c^2/2\sigma_{\text{prior}}^2)$ . Detection probability depends only on the sensor and terrain:

$$p_{\text{det}}(z) = \text{clip}(s_{\text{mode}}(z)(1-0.7f_c), 0.01, 0.99), \quad (1)$$

where  $s_{\text{mode}}(z)$  is the altitude-dependent detection probability (optical 0.29–0.97, thermal 0.97–0.99) and  $0.7f_c$  is canopy attenuation (up to 70%);  $p_{\text{det}}=0$  outside  $r_{\text{sense}}(z)$ . Here  $f_c \in [0, 1]$  is the local canopy-cover fraction; the clip to  $[0.01, 0.99]$  prevents the Bayesian update from freezing at certainty. Radio does not change what the camera detects; it determines whether the detected observation reaches the controller. Each uplink packet is independently lost with probability BLER (Table I). A lost packet provides no new information to the controller, which therefore reuses the most recent observation available for that drone. Equation (2) does not penalize packet loss directly; rather, packet loss reduces the number of detections that can contribute to belief accumulation, because observations that never reach the controller cannot influence its belief state. At harsh/extreme BLER ( $\geq 22.9\%$ ), belief accumulates too slowly for any cell to cross 0.5 within the mission, producing the radio hard limit (Sec. VI-B). With false-alarm rate  $p_{\text{fp}}=0.02$ , each received observation updates every cell  $c$  within footprint  $r_{\text{sense}}(z)$  by Bayes' rule:

$$B_{t+1}(c) = \frac{p B_t(c)}{p B_t(c) + q(1-B_t(c))}, \quad (2)$$

where  $(p, q) = (p_{\text{det}}(z), p_{\text{fp}})$  on a received detection, else  $(1 - p_{\text{det}}(z), 1 - p_{\text{fp}})$  on a received non-detection; a lost packet skips the sensor update (Eq. (2)). A cell confirms when its belief exceeds 0.5; victim mobility  $\sigma_{\text{mob}} = \bar{v} \Delta t_{\text{dec}}$  ( $\bar{v} \in [0, 0.7]$  m/s) diffuses the map every step regardless. Altitude  $z \in \{40, 80, 120\}$  m gives sensing radius  $r_{\text{sense}} = 500z/120$  m (167, 333, 500 m); flying at 120 m maximizes the footprint and  $s_{\text{mode}}(z)$  without sacrificing line of sight.

To prevent simultaneous recharge landings at FOB, initial charge levels are staggered, keeping  $\geq \frac{2}{3}$  ( $N=3$ ) or  $\geq \frac{5}{6}$  ( $N=6$ ) drones airborne throughout each mission.

### C. CPPO Swarm Controller

CPPO (CTCE) maintains a single joint policy centrally; no learned component resides on any drone. Unlike CTDE methods, all decisions travel through the LTE link, so channel impairments reach the training loop directly. At each step the controller sees, per drone: position  $c_t^i$ , altitude, detection flag  $d_t^i$ , battery, SINR, BLER, and RTT (7 scalars, normalized to  $[0, 1]$ ; lost uplinks substitute the most recent received observation, Sec. IV-A), plus the shared  $25 \times 25$  Bayesian belief map (Eq. (2)). A two-layer 256-unit network with Rectified Linear Unit (ReLU) and softmax heads, trained with PPO [23] and Generalized Advantage Estimation (GAE) [24] on a 28-node cluster [25], [26], outputs target cell  $w_t^i \in \{0 \dots 624\}$  and altitude  $\alpha_t^i \in \{0, 1, 2\}$  per drone, packed into downlink packets (Sec. IV-A). The reward has three goals: (1) detection ( $200\bar{D}_t$ , earned only from packets that survive the link; BLER thins the PPO gradient directly); (2) exploration ( $3\Delta C_t$ : new cells visited); and (3) information gain ( $8\Delta I_t$ : belief updates on non-detections). Penalties (8 per dead drone, 200 per lost victim, 0.004 per step) stop hovering:  $r_t = 200\bar{D}_t + 3\Delta C_t + 8\Delta I_t - \text{penalties}$ . Weights were tuned with Optuna [27] in preliminary ideal-LTE runs, so that no single term dominated, and then reused unchanged in all 66 scenarios; testing simpler versions is left for future work.

### V. EXPERIMENTAL METHODOLOGY

Training runs on a 28-node Linux cluster (Intel Core i7 desktops of four generations: 17 nodes with 4 cores/8 threads (i7-860/2600/4790) and 11 with 6 cores/12 threads (i7-8700); 2.8–3.6 GHz, 16 GB RAM each; 268 logical CPUs; gigabit Ethernet; CPU-only) using Ray; each worker runs one ns-3 LENA and one headless Godot 4 instance in lockstep, resetting after each episode.

**Experimental matrix.** The 66-run matrix comprises: 2 ideal baselines (perfect LTE, stationary victim); 48 LTE-severity runs (4 LTE severity levels, 6 LTE profiles, and 2 swarm sizes  $N \in \{3, 6\}$ ); and 16 time-of-day (TOD) configurations (4 time windows, 2 sensor modes, and 2 swarm sizes  $N \in \{3, 6\}$ , with ideal LTE and wandering behavior). Results average three seeds (2M PPO steps each) with fixed FOB, terrain, and hyperparameters. To investigate training performance under fixed conditions, in each of the 66 runs, an attempt is made to train a control policy using CPPO with the same initialization. Note that in each run, conditions are fixed; training performance under time-varying conditions is future work.

On this hardware each run averages 32 minutes ( $N=3$ ,  $\approx 1050$  steps/s) or 45 minutes ( $N=6$ ,  $\approx 735$  steps/s); fleet size, not channel severity, drives this gap (each varies under 5% by tier). The full 66-scenario, three-seed sweep totals roughly 127 compute-hours. Six victim-mobility profiles (Table III) follow Koester’s lost-person catalogue [22]; speed scales with time-of-day and slope, dropping 80% at night.

TABLE III

VICTIM-MOBILITY PROFILES AND PER-EPISEDE SAMPLING WEIGHTS FROM [22]. THE PROBABILITY COLUMN GIVES THE CHANCE EACH PROFILE IS DRAWN PER EPISEDE.

| Profile        | Base speed (m/s) | Probability |
|----------------|------------------|-------------|
| stationary     | 0.0              | 0.20        |
| wander         | 0.3              | 0.20        |
| linear_feature | 0.6              | 0.20        |
| straight_line  | 0.6              | 0.15        |
| climbing       | 0.5              | 0.10        |
| retrace        | 0.7              | 0.15        |

**Metrics.** Five outcomes per run: *discovery* (any cell belief  $> 0.5$ ), *coverage* (fraction of cells visited), *episode return*, *terminal belief mean squared error (MSE)*, and *LTE telemetry* (SINR, BLER, RTT); these raw radio measurements are reported per tier alongside RL outcomes.

### A. Non-Learning Controller Baseline (Lawnmower)

The lawnmower was chosen because its deterministic coverage isolates communication-induced learning failures without introducing optimization of its own: it is an IAMSAR (International Aeronautical and Maritime Search and Rescue) parallel-track sweep [20] that re-runs all 66 scenarios on the same testbed (LTE channel, sensor model, battery, stagger), dividing the  $25 \times 25$  grid into  $N$  strips with no inter-drone messages. The lawnmower flies at a fixed 120 m altitude to maximize its sensing footprint. Since CPPO also converges to 120 m in ideal cases (Table V), altitude does not explain the coverage gap in mild and medium tiers. Coverage is fixed (64.2%/85.6% at  $N=3/N=6$ ,  $\sigma=0$ ); sensor readings traverse the LTE link, subject to the same packet-loss rate (BLER, Table I), but the lawnmower has no gradient to thin.

### VI. RESULTS

### A. Network Performance Characterization

Table IV summarizes per-tier network statistics across 66 scenarios. Packet delivery ratio (PDR) drops from 0.999 (ideal) to 0.286 (extreme), and ZOH burst length grows  $240\times$ , wasting decision steps even when packets continue to arrive; at extreme LTE, radio link failures (RLF) cut off drones for seconds.

TABLE IV

NETWORK PERFORMANCE STATISTICS PER LTE TIER (POOLED  $N=3, 6$ ).  $\bar{L}_{\text{zoh}}$  IN 14 S SLOTS.

| Tier    | SINR (dB) | BLER (%) | RTT (ms) | $T_{\text{app}}$ (Mbps) | PDR   | $\bar{L}_{\text{zoh}}$ | RLF rate |
|---------|-----------|----------|----------|-------------------------|-------|------------------------|----------|
| ideal   | 34.9      | 0.10     | 0        | 98.1                    | 0.999 | 0.02                   | 0.0      |
| mild    | 23.0      | 3.96     | 64       | 14.1                    | 0.962 | 0.21                   | 0.0      |
| medium  | 14.7      | 10.7     | 94       | 8.7                     | 0.876 | 0.71                   | 0.1      |
| harsh   | 7.1       | 22.9     | 134      | 3.6                     | 0.624 | 2.34                   | 1.4      |
| extreme | -6.6      | 43.7     | 184      | 2.1                     | 0.286 | 4.81                   | 5.7      |

### B. Lawnmower vs. CPPO: Factoring the Cliff

Three regimes emerge from the per-tier comparison.

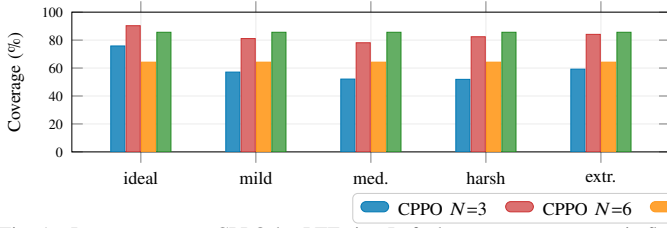


Fig. 1. Lawnmower vs. CPPO by LTE tier. Left: lawnmower coverage is flat ( $\sigma=0$ ); CPPO declines from mild onward (learnability cliff). Right: both reach 0% discovery at harsh/extreme (physical ceiling); CPPO leads at ideal LTE.

**Ideal LTE: RL wins.** Under a clean channel with no impairments, CPPO outperforms the lawnmower for both swarm sizes. With  $N=3$ , coverage rises 11.6% (75.8% vs. 64.2%) and discovery from 70% to 86%. With  $N=6$ , coverage gains 4.7% (90.3% vs. 85.6%) and discovery from 75% to 92%. CPPO steers drones toward high-probability areas and spreads them to avoid overlap.

**Mild/medium LTE: RL falls behind.** CPPO drops to 57.1%/81.1% (mild) and 52.1%/78.1% (medium) for  $N=3/N=6$ , a 4.5–12% gap below the lawnmower (64.2%/85.6%,  $\sigma=0$ ; left panel of Fig. 1). Coverage needs no reward signal, so the gap is a pure learning problem: fewer observations reach the controller as the channel degrades, and ZOH bursts (Sec. VI-C) waste 14-second decision steps. The lawnmower has no gradient and is unaffected.

**Harsh/extreme: radio hard limit.** Both policies reach 0% discovery; at BLER  $\geq 22.9\%$ , each lost uplink withholds one Bayesian update, so observations arrive too sparsely to push any cell's belief above the 0.5 confirmation threshold within the mission window, and ZOH bursts ( $\bar{L}_{\text{zoh}} \geq 2.34$  slots) further reduce effective decision steps.

Mann-Whitney tests [28] confirm perfect rank separation for mild/medium tiers ( $U=0$ ,  $p=0.05$ , Cohen's  $d>1.4$ ); harsh/extreme produce zero variance.

Even at extreme LTE, coverage holds at 52–86% as ZOH replays the last waypoint, yet no cell confirms because each lost packet withholds a Bayesian update (Sec. IV-B). The transition from SINR = 14.7 dB / BLER = 10.7% (medium) to SINR = 7.1 dB / BLER = 22.9% (harsh) marks the physical boundary between the learnability cliff and the radio hard limit.

### C. Failure Mode Breakdown

Three mechanisms explain the mild/medium gap. (i) **Sparse reward:** fewer arriving packets mean fewer useful reward events per step, leaving the policy undertrained within the 2 M-step budget. Fleet size compounds this:  $N=6$  multiplies uplink traffic and contention, so per-drone delivery rates fall faster than at  $N=3$ . (ii) **Stale commands:**  $\bar{L}_{\text{zoh}}$  grows from 0.21 (mild) to 0.71 (medium) slots (Table IV), wasting 14-second decision steps; the lawnmower advances its waypoint regardless, so its coverage is unaffected. (iii) **Hard confirmation floor:** once BLER exceeds 22.9% (harsh tier), observations arrive too sparsely for any cell to cross the 0.5 threshold within the mission (Sec. IV-B). Fig. 2 (left) shows discovery dropping from 67%/83% ( $N=3/N=6$ ; mild) to 28%/15% (medium) and collapsing at harsh; the right panel traces BLER and RTT trajectories (64–184 ms) driving the collapse.

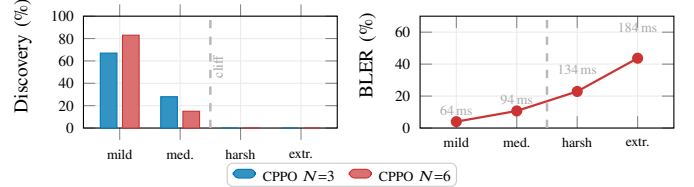
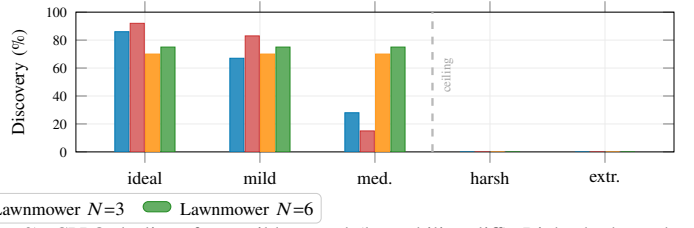


Fig. 2. Failure mode breakdown: CPPO victim discovery by swarm size (left) and BLER with RTT annotations (right) vs. LTE tier. Discovery collapses between medium and harsh where BLER exceeds 22%.

### D. Learning Dynamics and the Learnability Cliff

The right panel of Fig. 1 shows final discovery at the 2 M-step mark. Ideal LTE converges fastest ( $N=6$ : 92% by 1.8 M;  $N=3$ : 86% by 1.45 M). Mild LTE plateaus at 67%/83% ( $N=3/N=6$ ), below the lawnmower, marking the learnability cliff. Medium LTE stalls near 15%/28% and never improves. Harsh and extreme remain at 0% throughout: at BLER  $\geq 22.9\%$ , no cell reaches the 0.5 confirmation threshold within the mission (Sec. IV-B), regardless of flight strategy.

### E. What the Agent Actually Learns Under Ideal LTE

Fig. 3 shows flight-path evolution under ideal LTE for  $N=6$ , defining the target behavior that radio degradation prevents. Drones start at the FOB (■, center); the prior centers on the a priori position (◆, SE corner), while the victim (★) is in the NW corner, a deliberate worst case for the prior. The left panel (early training) crowds the SE quadrant; the right panel (converged) fans out uniformly (92% discovery, 90.3% coverage, +4.7% over lawnmower), which is the behavior radio degradation prevents in all other tiers.

### F. Observed Differences Are Consistent With Radio Effects

Table V holds all non-radio variables (sensor type, battery, terrain, starting position, victim location, and training seed) fixed across tiers. Battery level, swap count, and flight time remain nearly constant across channel conditions; even at harsh LTE, the  $N=6$  swarm flies at 9.3 m/s yet finds zero victims. Among all measured metrics, only discovery tracks channel quality; altitude, speed, battery, and swap count are identical across tiers. This pattern is consistent with channel quality as the driving factor: at extreme LTE (BLER 43.7%), only 56.3% of observations reach the controller, starving the belief map (Sec. IV-B); the lawnmower's fixed coverage ( $\sigma=0$ , Sec. V-A) confirms flight geometry is unchanged across tiers.

The staggered launch keeps  $\geq 97\%$  of the  $N=6$  swarm airborne; without it, availability collapses to 3.8 drones and costs 8% of discovery even under ideal LTE.

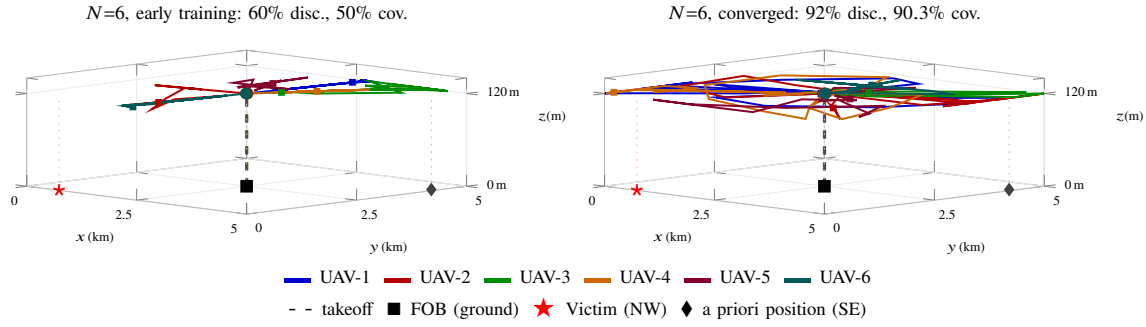


Fig. 3. 3-D SAR flight paths under ideal LTE ( $z=120$  m cruise altitude),  $N=6$ . *Left*: early training (58 K steps); *right*: converged (2M steps, 92% discovery, 90.3% coverage). The victim (NW) lies opposite the a priori position (SE): a deliberate worst case, with placement randomized across the 66-scenario suite.

TABLE V

FLIGHT/ENERGY INVARIANTS ( $\bar{z}$ : MEAN ALTITUDE,  $f_{120}$ : FRACTION OF STEPS AT 120 M MAX ALTITUDE,  $\bar{b}$ : MEAN BATTERY FRACTION AT EPISODE END). ONLY DISCOVERY MOVES WITH THE LTE CLIFF; ALTITUDE, SPEED, BATTERY, AND SWAP COUNT ARE CHANNEL-INVARIANT.

| LTE   | $\bar{z}$ (m) |     | $f_{120}$ |      | $\bar{v}$ (m/s) |      | $\bar{b}$ |      | swaps/run |     | disc (%) |     |
|-------|---------------|-----|-----------|------|-----------------|------|-----------|------|-----------|-----|----------|-----|
|       | N=3           | N=6 | N=3       | N=6  | N=3             | N=6  | N=3       | N=6  | N=3       | N=6 | N=3      | N=6 |
| ideal | 120           | 120 | 1.00      | 1.00 | 12.7            | 10.4 | 0.51      | 0.51 | 8         | 15  | 86       | 92  |
| mild  | 120           | 120 | 1.00      | 1.00 | 13.4            | 11.1 | 0.52      | 0.51 | 8         | 16  | 67       | 83  |
| harsh | 120           | 120 | 1.00      | 1.00 | 5.1             | 9.3  | 0.51      | 0.50 | 6         | 13  | 0        | 0   |

### G. Discovery vs. Average SINR

Fig. 4 plots victim discovery against the per-tier average SINR for CPPO and the lawnmower at both swarm sizes, tracing the two failure regimes step by step from left (worst channel) to right (best).

**SINR  $\leq 7.1$  dB (harsh/extreme): radio hard limit.** Discovery is zero for every controller (Sec. VI-B): no flight strategy survives this radio and mission-duration ceiling.

**SINR = 14.7 dB (medium): learnability cliff.** The lawnmower holds at 70%/75% discovery ( $N=3/N=6$ ) because its fixed sweep is unaffected by reward sparsity. CPPO falls to 28%/15% (a 42–60% drop) even though the channel still delivers packets. BLER at 10.7% and RTT at 94 ms hollow out the training signal without silencing the link: commands arrive, but too few reward events reach the trainer per rollout to improve the policy; only 89 of every 100 reach the trainer, compounded by ZOH freezing commands for 0.71 slots on average.

**SINR  $\geq 23.0$  dB (mild/ideal): learnable.** Both CPPO swarm sizes reach or exceed the lawnmower. The gradient is sufficiently dense to exploit sensor detections; CPPO’s advantage over the lawnmower (+11.6% at  $N=3$ , +4.7% at  $N=6$  under ideal LTE) comes from directing drones toward high-probability cells rather than sweeping blindly.

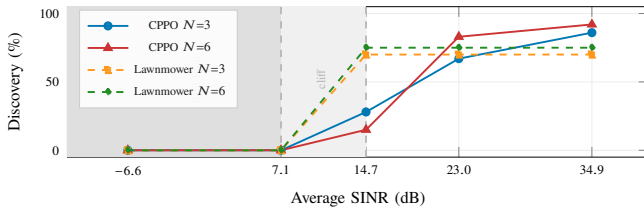


Fig. 4. Discovery vs. average SINR per tier. Dark gray ( $\leq 7.1$  dB): radio hard limit, zero discovery for every controller. Light gray (7.1–14.7 dB): learnability cliff, CPPO collapses while the lawnmower holds. Above 23.0 dB both controllers succeed and CPPO leads.

Coverage does not explain the discovery gap (Sec. VII): the

SINR axis alone identifies both failure boundaries, without any composite score.

**The learning loss.** On a given channel, physical detection probability  $p_{\text{det}}$  (Eq. (1)) and packet-loss rate (BLER, Table I) are identical for both controllers, so the lawnmower and CPPO face the same per-overflight chance of a detection reaching the controller. Yet in mild and medium LTE, CPPO discovers fewer victims. That gap is a learning failure, not a radio limit, and it is the central finding of this paper. This finding motivates channel-aware training schemes (reward shaping, curriculum scheduling, or adaptive rollout length [8]), which are relevant only once the learning failure has been confirmed to exist independently of the radio ceiling.

## VII. DISCUSSION

**Two failure modes.** LTE degradation hurts the swarm in two different ways, and each calls for different fixes. At harsh and extreme quality the radio itself is the wall: so few detections survive the link that no controller, learning or not, can confirm a victim. At mild and medium quality, the link still delivers enough packets for the lawnmower to work, yet CPPO falls 4.5–12% in coverage behind it. Here the problem is not the radio, but learning: the reward signal has become too sparse to train the policy. Telling the two modes apart matters: a hard limit calls for a better radio or a different mission, while a learnability cliff calls for changes to training such as curriculum scheduling or reward shaping [8]. Without this baseline, CPPO’s 28%/15% discovery under medium LTE invites a hardware upgrade, when the real bottleneck is a thinned training signal that a revised curriculum or reward scheme could overcome at the same link quality.

**The lawnmower baseline is a transferable method.** Any RL system whose training signal travels over a wireless link (industrial control, robotics, delivery drones) can run a non-learning controller on the same channel to separate radio limits from learning limits before committing to a fix. The comparison is most informative when the non-RL controller matches the RL agent in coverage, so any gap isolates the learning deficit from flight geometry. The lawnmower’s fixed sweep needs no training and cannot itself suffer a learnability failure; other untested baselines (random search, frontier exploration) could further test the cliff’s generality.

**Why packet loss and stale commands hurt learning.** Lost uplinks and downlinks (Sec. IV-A) keep drones flying but

produce no fresh reward for training. As the channel worsens these gaps grow, so fewer rollout steps carry useful signal, and longer training helps little: halving useful steps roughly doubles the channel quality needed to learn. This is why mild and medium LTE plateau below the lawnmower instead of slowly catching up. Fleet scaling compounds the effect: at  $N=6$ , six concurrent uplink streams share the same 50 resource blocks, driving per-drone delivery below the per-tier mean.

**Coverage alone misleads** (Sec. VI-G): it hides both the hard limit (52–86% covered, yet zero victims at harsh/extreme) and the mid-tier gap (medium,  $N=3$ : CPPO 52% vs. lawnmower 64%). Studies reporting only coverage [16] miss radio-induced failures. Average SINR, reported directly per tier, cleanly separates both regimes without requiring a derived composite.

### VIII. CONCLUSION

LTE impairments impose two distinct failure modes on RL-controlled swarms: a radio hard limit (harsh/extreme) that blocks all controllers equally, and an RL learnability cliff (mild/medium) where CPPO falls 4.5–12% in coverage below the lawnmower because its reward signal is too sparse to train on. Running the lawnmower baseline on the same channel is what makes the two separable, and the method transfers to any wireless RL system. Plotting discovery against average SINR per tier (Fig. 4) traces both thresholds directly, giving a rule of thumb for this testbed: below 7.1 dB no controller discovers; between 7.1 and 23.0 dB the fix belongs in training, not the radio; above 23.0 dB the task is fully learnable, with RL gaining 4.7–11.6% in coverage over the lawnmower sweep.

**Limitations and future work.** This is Phase 1: a simulation study using three seeds across a structured scenario suite. The reported thresholds (7.1, 14.7, 23.0 dB) are therefore indicative rather than universal: they depend on the episode length, victim mobility, terrain, and swarm size used here, and will change as these parameters vary, as well as with victim count, sensor modality, and channel model (rural LTE here; urban/5G-NR links differ in fading and access). Each policy trained under one fixed condition; future work will test robustness to LTE severity varying during training. Phase 2 deploys the pretrained policy on hardware to evaluate whether the SINR-based thresholds generalize out-of-suite without retraining, and extends the framework to 5G/NR admission, SINR-adaptive scheduling, and reward shaping to move the learnability cliff into more challenging channel conditions.

**Reproducibility.** The ns-3 LENA configuration, PPO training scripts, Godot environment, and all per-scenario result logs are archived at <https://github.com/Nakok/Search-and-Rescue>.

### REFERENCES

- [1] P. Ameigeiras, J. Navarro-Ortiz, P. Andres-Maldonado, J. M. Lopez-Soler, J. Lorca, Q. Perez-Tarrero, and R. Garcia-Perez, “3GPP QoS-based scheduling framework for LTE,” *EURASIP J. Wireless Commun. Netw.*, 2016, art. no. 78.
- [2] M. M. Azari, F. Rosas, K.-C. Chen, and S. Pollin, “Ultra reliable UAV communication using altitude and cooperation diversity,” *IEEE Trans. Commun.*, vol. 66, no. 1, pp. 330–344, 2018.
- [3] X. Lin, V. Yajnanarayana, S. D. Muruganathan, S. Gao, H. Asplund, H.-L. Mänttinen, M. Bergström, S. Euler, and Y.-P. E. Wang, “The sky is not the limit: LTE for unmanned aerial vehicles,” *IEEE Commun. Mag.*, vol. 56, no. 4, pp. 204–210, 2018.
- [4] T. Rashid, M. Samvelyan, C. S. de Witt, G. Farquhar, J. N. Foerster, and S. Whiteson, “QMIX: Monotonic value function factorisation for deep multi-agent reinforcement learning,” in *Proc. 35th Int. Conf. Machine Learning (ICML)*, Stockholm, Sweden, 2018, pp. 4295–4304.
- [5] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch, “Multi-agent actor-critic for mixed cooperative-competitive environments,” in *Advances in Neural Information Processing Systems (NeurIPS)*, Long Beach, CA, 2017, pp. 6382–6393.
- [6] C. Yu, A. Velu, E. Vinitsky, J. Gao, Y. Wang, A. Bayen, and Y. Wu, “The surprising effectiveness of PPO in cooperative, multi-agent games,” in *Advances in Neural Information Processing Systems (NeurIPS)*, New Orleans, LA, 2022, pp. 24 611–24 624.
- [7] M. Wen, J. G. Kuba, R. Lin, W. Zhang, Y. Wen, J. Wang, and Y. Yang, “Multi-agent reinforcement learning is a sequence modeling problem,” in *Advances in Neural Information Processing Systems (NeurIPS)*, New Orleans, LA, 2022, pp. 16 509–16 521.
- [8] M. Andrychowicz, A. Raichuk, P. Stańczyk, M. Orsini, S. Girgin, R. Marinier, L. Hussenot, M. Geist, O. Pietquin, M. Michalski, S. Gelly, and O. Bachem, “What matters for on-policy deep actor-critic methods? A large-scale study,” in *Int. Conf. Learning Representations (ICLR)*, Online, 2021.
- [9] P. Henderson, R. Islam, P. Bachman, J. Pineau, D. Precup, and D. Meger, “Deep reinforcement learning that matters,” in *Proc. AAAI Conf. Artificial Intelligence*, New Orleans, LA, 2018, pp. 3207–3214.
- [10] S. Sukhbaatar, A. Szlam, and R. Fergus, “Learning multiagent communication with backpropagation,” in *Advances in Neural Information Processing Systems (NeurIPS)*, Barcelona, Spain, 2016, pp. 2244–2252.
- [11] J. N. Foerster, Y. M. Assael, N. de Freitas, and S. Whiteson, “Learning to communicate with deep multi-agent reinforcement learning,” in *Advances in Neural Information Processing Systems (NeurIPS)*, Barcelona, Spain, 2016, pp. 2137–2145.
- [12] A. Singh, T. Jain, and S. Sukhbaatar, “Learning when to communicate at scale in multiagent cooperative and competitive tasks,” in *Int. Conf. Learning Representations (ICLR)*, New Orleans, LA, 2019.
- [13] A. Das, T. Gervet, J. Romoff, D. Batra, D. Parikh, M. Rabbat, and J. Pineau, “TarMAC: Targeted multi-agent communication,” in *Proc. 36th Int. Conf. Machine Learning (ICML)*, Long Beach, CA, 2019, pp. 1538–1546.
- [14] R. Wang, X. He, R. Yu, W. Qiu, B. An, and Z. Rabinovich, “Learning efficient multi-agent communication: An information bottleneck approach,” in *Proc. 37th Int. Conf. Machine Learning (ICML)*, Online, 2020, pp. 9908–9918.
- [15] Y. F. Chen, M. Liu, M. Everett, and J. P. How, “Decentralized non-communicating multiagent collision avoidance with deep reinforcement learning,” in *IEEE Int. Conf. Robotics and Automation (ICRA)*, Singapore, 2017, pp. 285–292.
- [16] Z. Mou, Y. Zhang, F. Gao, H. Wang, T. Zhang, and Z. Han, “Deep reinforcement learning based three-dimensional area coverage with UAV swarm,” *IEEE J. Sel. Areas Commun.*, vol. 39, no. 10, pp. 3160–3176, 2021.
- [17] W. Khawaja, I. Guvenc, D. W. Matolak, U.-C. Fiebig, and N. Schneckenburger, “A survey of air-to-ground propagation channel modeling for unmanned aerial vehicles,” *IEEE Commun. Surv. Tutor.*, vol. 21, no. 3, pp. 2361–2391, 2019.
- [18] Q. Wu, J. Xu, Y. Zeng, D. W. K. Ng, N. Al-Dhahir, R. Schober, and A. L. Swindlehurst, “A comprehensive overview on 5G-and-beyond networks with UAVs: From communications to sensing and intelligence,” *IEEE J. Sel. Areas Commun.*, vol. 39, no. 10, pp. 2912–2945, 2021.
- [19] S. Odero, D. Makaroff, D. Eager, and C. Tamme, “Remote reinforcement learning for UAVs,” in *Proc. 13th EAI Int. Conf. Mobility, IoT and Smart Cities (EAI MobilityIoT)*, Online, 2026, pp. 1–20.
- [20] International Maritime Organization and International Civil Aviation Organization, *IAMSAR Manual, Volume II: Mission Co-ordination*, 2019th ed. London/Montreal: IMO/ICAO, 2019.
- [21] N. Baldo, M. Miozzo, M. Requena-Esteso, and J. Nin-Guerrero, “An open source product-oriented LTE network simulator based on ns-3,” in *Proc. 14th ACM Int. Conf. Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*, Miami, FL, 2011, pp. 293–298.
- [22] R. J. Koester, *Lost Person Behavior: A Search and Rescue Guide on Where to Look*. Charlottesville, VA: dbS Productions, 2008.
- [23] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, “Proximal policy optimization algorithms,” 2017.
- [24] J. Schulman, P. Moritz, S. Levine, M. I. Jordan, and P. Abbeel, “High-dimensional continuous control using generalized advantage estimation,” in *Proc. 4th Int. Conf. Learning Representations (ICLR)*, San Juan, Puerto Rico, 2016.
- [25] A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, and N. Dormann, “Stable-baselines3: Reliable reinforcement learning implementations,” *Journal of Machine Learning Research*, vol. 22, no. 268, pp. 1–8, 2021.
- [26] P. Moritz, R. Nishihara, S. Wang, A. Tumanov, R. Liaw, E. Liang, M. Elibol, Z. Yang, W. Paul, M. I. Jordan, and I. Stoica, “Ray: A distributed framework for emerging AI applications,” in *Proc. 13th USENIX Symp. Operating Systems Design and Implementation (OSDI)*, Carlsbad, CA, 2018, pp. 561–577.
- [27] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, “Optuna: A next-generation hyperparameter optimization framework,” in *Proc. 25th ACM SIGKDD Int. Conf. Knowledge Discovery & Data Mining (KDD)*, Anchorage, AK, 2019, pp. 2623–2631.
- [28] T. W. MacFarland and J. M. Yates, “Mann–whitney u test,” in *Introduction to nonparametric statistics for the biological sciences using R*. Cham, Switzerland: Springer, 2016, pp. 103–132.