electronics

MDPI

Article

CSOOC: Communication-State Driven Online–Offline Coordination Strategy for UAV Swarm Multi-Target Tracking

Haoran Sun [1], Yicheng Yan [1], Guojie Liu [2], Ying Zhan [1] and Xianfeng Li [1,*]

[1] The School of Computer Science and Engineering, Macau University of Science and Technology, Taipa, Macau, China; 2202853ein30001@student.must.edu.mo (H.S.); 21098535in30001@student.must.edu.mo (Y.Y.); 2009853mia30001@student.must.edu.mo (Y.Z.)

[2] The School of Cyber Science and Engineering, Zhengzhou University, Zhengzhou 450001, China; lgjzzu@zzu.edu.cn

* Correspondence: xifli@must.edu.mo

Abstract

Unmanned aerial vehicle (UAV) swarms have shown great potential in large-scale IoT (Internet of Things) and smart agriculture applications, particularly for cooperative monitoring and multi-target tracking in field environments. However, most existing coordination strategies assume ideal communication conditions, overlooking realistic network impairments such as congestion, packet loss, and latency. These impairments disrupt the timely exchange of information between UAVs and the ground base station, leading to delayed or lost control signals. As a result, coordination quality deteriorates and tracking performance is severely degraded in real-world deployments. To address this gap, we propose CSOOC (Communication-State Driven Online–Offline Coordination with Congestion Control), a hybrid control architecture that integrates centralized learning-based decision-making with decentralized rule-based policies to adapt UAV behaviors according to real-time network states. CSOOC consists of three key components: (1) an online module that enables centralized coordination under reliable communication, (2) an offline profit-driven mobility strategy based on local Gaussian maps for autonomous target tracking during communication loss, and (3) a congestion control mechanism based on STAR(Stratified Transmission and RTS/CTS), which combines temporal transmission desynchronization and RTS/CTS handshaking to enhance uplink reliability. We establish a unified co-simulation paradigm that connects network communication with swarm control and swarm coordination behavior. Experiments demonstrate that CSOOC achieves an average observation rate of 39.7%, surpassing baseline algorithms by 4.4–11.13%, while simultaneously improving network stability through significantly higher packet delivery ratios under congested conditions. These results demonstrate that CSOOC effectively bridges the gap between algorithmic performance in simulation and practical UAV swarm operations in communication-constrained environments.

Keywords: UAVs; multi-target tracking; communication-aware; congestion control

check for updates

Academic Editor: Stefano Scanzio

Received: 12 November 2025
Revised: 29 November 2025
Accepted: 1 December 2025
Published: 2 December 2025

Citation: Sun, H.; Yan, Y.; Liu, G.; Zhan, Y.; Li, X. CSOOC: Communication-State Driven Online–Offline Coordination Strategy for UAV Swarm Multi-Target Tracking. *Electronics* **2025**, *14*, 4743. https://doi.org/10.3390/electronics14234743

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

In recent years, UAV swarms have demonstrated remarkable potential in large-scale IoT and smart agriculture applications, such as crop monitoring, precision spraying, and environmental sensing [1–3]. A key capability underlying these applications is the cooperative observation of multiple moving or dynamic targets in complex environments—formally defined as the Cooperative Multi-Robot Observation of Multiple Moving Targets (CMOMMT)

problem [4]. The central objective of CMOMMT is to maximize the average observation rate across all targets by continuously balancing target following and exploratory search, despite the limited field-of-view and mobility constraints of each UAV, as illustrated in Figure 1.

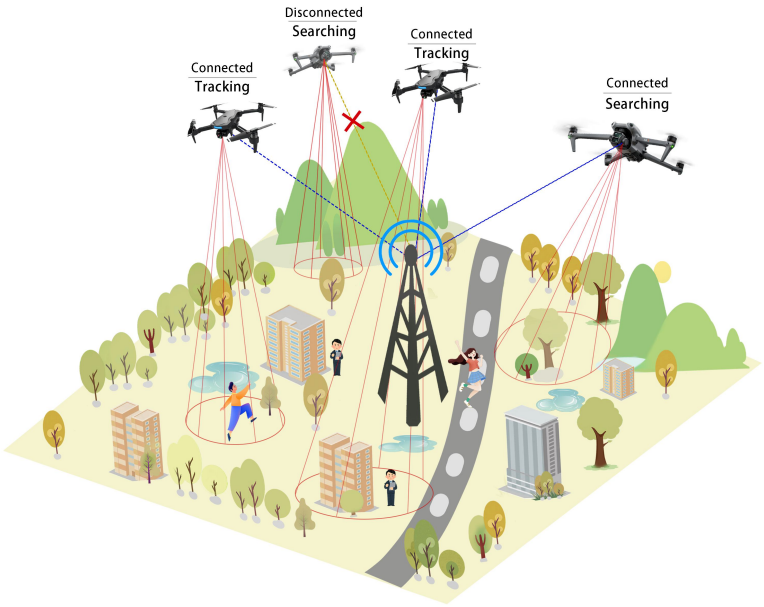


Figure 1. Illustration of a UAV swarm performing cooperative target tracking in an IoT sensing scenario. Communication congestion between multiple UAVs and the GBS can lead to packet loss and delayed control feedback.

A wide spectrum of coordination strategies has been explored to address the CMOMMT problem, ranging from classical heuristic methods to modern learning-based frameworks [4–8]. Early approaches, such as force-vector and frontier-based methods, relied on local rules for decentralized cooperation. Later, optimization-based strategies (e.g., profit-driven algorithms, Particle Swarm Optimization, and Ant Colony Optimization) improved task allocation efficiency but required handcrafted objectives. Reinforcement learning (RL) methods further enhanced adaptiveness but often suffered from high sample complexity and instability in large-scale scenarios.

Among learning-based approaches, our previous work proposed the Supervised and Transfer Learning Framework (STLF) [9], a two-stage framework that trains a Teacher Model with privileged global information and transfers knowledge to a Student Model operating under partial observability. STLF demonstrated strong performance in achieving high observation rates and outperforming several baselines. However, the original STLF operates under an idealized assumption of reliable and synchronous communication between UAVs and the Ground Base Station (GBS). In practice, UAV swarms often operate in wireless environments with limited bandwidth and shared uplink channels. Building upon the original STLF framework, this work enhances its applicability to realistic network environments by incorporating mechanisms to handle communication impairments such as congestion, collisions, and packet loss, which are frequently encountered in real-world UAV swarm deployments.

When multiple UAVs simultaneously transmit status updates or receive commands from the GBS, network congestion can occur, leading to packet loss, delayed feedback, or even complete link failures [10,11]. Since centralized policies like STLF depend on timely and reliable communication, these network degradations can directly reduce coordination efficiency and overall tracking performance. Studies show that Ultra-Wideband can support accurate inter-UAV ranging, but even wideband systems remain constrained by limited

spectrum and sensitivity to interference in dense swarms [12,13]. These observations further highlight that communication quality is a fundamental bottleneck, motivating coordination strategies that explicitly account for communication states. This challenge calls for a communication-aware swarm coordination framework that can adapt UAV behaviors to the underlying network state in real time.

To address this challenge, we propose CSOOC, a hybrid control architecture that integrates learning-based and rule-based policies while explicitly accounting for communication dynamics. CSOOC consists of three tightly coupled modules. First, during online mode, UAVs execute globally coordinated control commands generated by the STLF model, leveraging centralized decision-making for efficient target observation. Second, during offline mode, UAVs switch to a profit-driven mobility strategy based on locally constructed Gaussian maps, enabling autonomous target following and exploration when communication is degraded. Finally, to ensure smooth and stable coordination between these two modes, a STAR congestion control module is integrated into the framework. By combining temporal transmission desynchronization with RTS/CTS-based (Request-To-Send/Clear-To-Send) [14] channel reservation, STAR mitigates network congestion and stabilizes uplink reliability, reducing unnecessary mode switching and maintaining consistent control performance.

The main contributions of this study are summarized as follows:

- A communication-aware UAV swarm coordination framework, termed CSOOC, is introduced to enable dynamic online–offline behavioral switching under realistic and time-varying network conditions.
- A profit-driven offline mobility strategy based on Gaussian maps is designed to maintain spatially informed target tracking during communication interruptions.
- A dual-layer congestion control mechanism, named STAR, is developed by combining temporal transmission stratification with RTS/CTS handshake to enhance network reliability and mitigate packet collisions.
- A unified co-simulation paradigm is established to bridge network communication and swarm control, allowing systematic evaluation of coordination performance under practical communication constraints.

These contributions also provide a new perspective on coupling network simulation with swarm coordination, enabling the study of UAV behavior under communication constraints. STAR reduces the impact of congestion, and the online–offline mechanism enables the swarm to adjust its coordination strategy according to current communication states.

The remainder of this paper is organized as follows. Section 2 reviews related work on CMOMMT coordination and communication-aware swarm control. Section 3 formulates the problem and system architecture. Section 4 presents the proposed CSOOC. Section 5 reports extensive experimental results under varying network and swarm configurations. Section 6 concludes the paper and discusses future directions.

2. Related Work

This section reviews related research from four perspectives, moving from UAV swarm coordination in CMOMMT tasks to communication constraints, network simulation, and congestion control.

2.1. UAV Swarm Coordination for CMOMMT Tasks

The CMOMMT problem [4] seeks to maximize persistent target observation via coordinated multi-agent behaviors. In UAV swarm scenarios, early methods such as B-CMOMMT [15] and P-CMOMMT [16] adopted heuristic and profit-driven task allocation, balancing tracking and exploration but often yielding suboptimal global performance.

Subsequent works introduced flexible formation control [17] and motion prediction [5] to improve coverage efficiency.

Recent studies applied Deep Reinforcement Learning (DRL) [18] to learn adaptive policies in complex environments, yet they typically assume stable communication. Our prior STLF [9] framework achieved high observation rates under ideal connectivity but did not address intermittent links. This gap motivates our integration of communication-aware behavior switching for robust UAV swarm coordination under both connected and disconnected states.

2.2. Communication-Constrained Multi-Agent Systems

Communication constraints critically influence coordination in multi-agent systems. Early models often assumed idealized connectivity [19], overlooking bandwidth, delay, and channel variability. Recent studies integrate communication awareness into planning, such as map compression for efficient online path planning [20]. In underwater AUV tasks, vision-based deep reinforcement learning enables joint motion and communication optimization in challenging acoustic channels [19]. For large-scale IoT and sensor networks, energy-aware and topology-adaptive strategies improve connectivity under limited resources [21,22]. These approaches demonstrate that coupling motion planning with realistic communication models is essential for robust performance in bandwidth- and latency-limited environments.

2.3. Network Simulation and Communication

Accurate evaluation of UAV communication strategies requires realistic network simulation frameworks capable of modeling channel dynamics, mobility, and protocol behavior. The NS-3 [23] simulator has been widely adopted for wireless network research due to its modular architecture and extensibility. The development of LTE and 5G modules for NS-3 has enabled researchers to assess cellular-based UAV communication under realistic PHY/MAC-layer conditions, incorporating mobility and interference effects [24,25]. Beyond protocol-specific modules, generic simulation platforms such as those surveyed by Jiang et al. [26] provide insights into selecting appropriate testbeds for UAV scenarios. Advanced simulation environments also allow integration with mobility simulators, enabling synchronized evaluation of both communication quality and UAV mission performance [27]. These tools facilitate the study of network performance in diverse conditions, from dense urban areas to open fields, supporting protocol optimization for UAV swarms. Recent works highlight the necessity of incorporating realistic propagation models, congestion effects, and mobility traces to bridge the gap between simulation and real-world deployments [10].

2.4. Congestion Control in UAV-Ground Communication

Efficient congestion control is critical for UAV-ground communications due to the dynamic topology, limited bandwidth, and high mobility of UAV networks. Cross-layer designs, such as the Mobility and Congestion-Aware Routing Protocol (MCARP), integrate routing and transport-layer mechanisms to adaptively adjust data transmission based on mobility prediction and link quality, thereby reducing packet loss and delay in dynamic UAV networks [28]. In TCP-based approaches, deep reinforcement learning has been leveraged to optimize congestion window adjustments in UAV-assisted wireless networks, outperforming traditional TCP variants under fluctuating link conditions [29]. Energy-aware methods further combine congestion control with data aggregation strategies to balance network throughput and UAV energy consumption, extending mission duration in multi-UAV-enabled IoT systems [30]. Opportunistic transmission schemes, such as those proposed in load-adaptive multi-hop routing, employ queue-length and channel state

information to prevent bottlenecks and improve delivery rates under heavy traffic [31]. Additionally, congestion-aware aerial-terrestrial hybrid architectures dynamically allocate resources between UAV relays and ground stations to mitigate interference and maintain QoS in heterogeneous environments [32]. Collectively, these studies highlight that combining congestion awareness with mobility prediction, cross-layer optimization, and energy efficiency is essential for sustaining robust UAV-ground communication under high-load scenarios.

3. Task Formulation

3.1. Mission Overview and Notation

In this study, we focus on the CMOMMT problem while incorporating realistic network communication modeling. By embedding a network simulator (NS-3 [23]) into the mission workflow, the decision-making process accounts for communication latency, packet loss, and bandwidth constraints, thus reflecting more practical UAV's operation scenarios.

The illustration of CMOMMT formulation is shown in Figure 2, and the key elements are defined as follows:

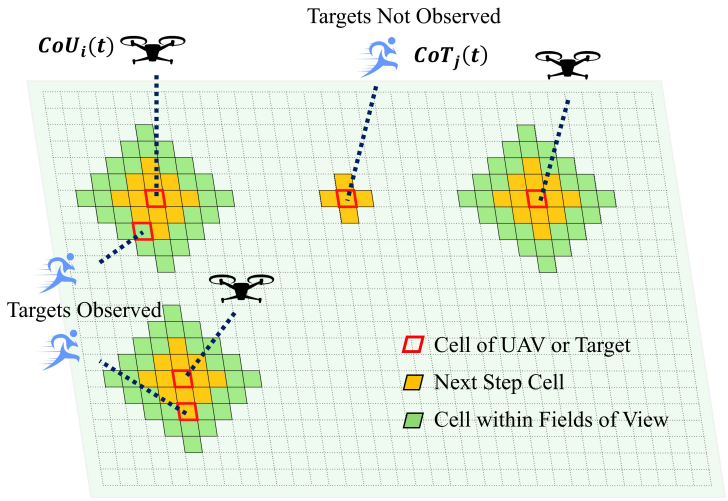


Figure 2. CMOMMT formulation [9].

Search Region. The search region is modeled as a 2D square grid, discretized into uniform cells represented by gray dashed lines.

Time Step. Time is discretized into steps indexed by $t = 1, 2, \ldots, T$, where T denotes the total mission duration.

Targets. There are M homogeneous targets. The position of target j at time step t is denoted by $CoT_j(t)$, indicating the grid cell it occupies. At each time step, targets move to one of the four adjacent cells (up, down, left, or right), following a bounded mobility model.

UAVs. A set of N homogeneous UAVs are deployed in the search region to observe the M targets ($N < M$). The UAVs have a higher maximum speed than the targets; otherwise, a target could easily evade pursuit by moving at its maximum speed. Additionally, UAVs maintain a fixed altitude throughout the mission. Each UAV is equipped with:

1. GPS for self-localization.
2. Onboard cameras to observe ground targets. UAV i observes cells within a given Euclidean radius, forming its field of view $FoV_i(t)$. If target j lies within $FoV_i(t)$, i.e., $CoT_j(t) \in FoV_i(t)$, it is considered observed. Areas outside $FoV_i(t)$ remain unknown to UAV i, reflecting partial observability.

3. Wireless communication modules to transmit observations to the GBS over a modeled network (e.g., LTE/5G) in NS-3, where transmission is subject to latency, packet loss, and congestion effects.

CMOMMT Workflow. Figure 3 shows the mission workflow. At each time step, UAVs first conduct local observations. The collected data are then transmitted via the simulated network to the GBS. The GBS aggregates information from all UAVs and computes coordinated movement decisions, which are sent back to the UAVs through the same network. This process, subject to communication delays and potential link interruptions, repeats until $t = T$.

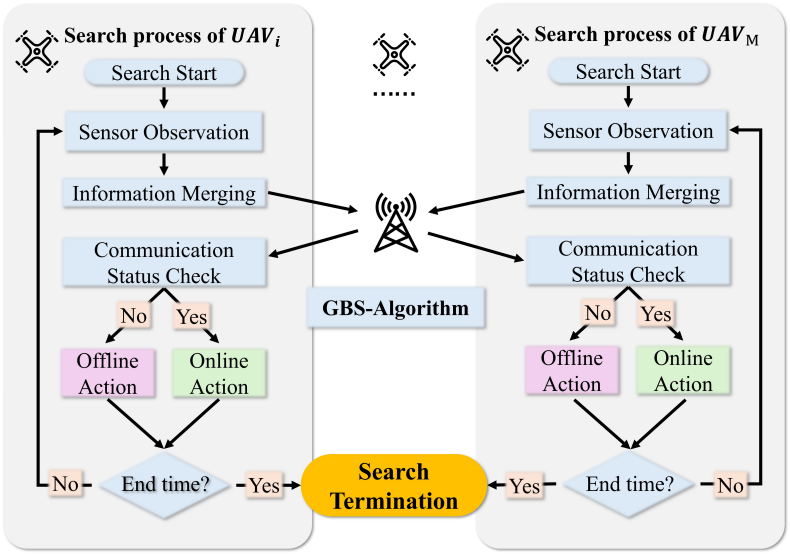


Figure 3. CMOMMT Workflow with integrated network simulation.

Objective Function. The objective is to maximize the average observation rate of all targets over T, denotedas $\Theta(T)$:

$$\Theta(T) = \frac{1}{M * T} \sum_{t=1}^{T} \sum_{j=1}^{M} O_j(t) \tag{1}$$

where $O_j(t)$ is a binary indicator of whether target j is observed at time t:

$$O_j(t) = \begin{cases} 1, & \text{if } \exists UAV_i : CoT_j(t) \in FoV_i(t) \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

In the ideal case where every target is observed at each time step, $\Theta(T) = 1$. However, due to UAV mobility constraints, limited FoV, and the effects of network latency and packet loss, achieving $O_j(t) = 1$ for all j, t is generally infeasible. The mission goal is to design a coordination policy that maximizes $\Theta(T)$ under both mobility and communication constraints.

3.2. Target Mobility Model

To construct a biomimetic target trajectory, we adopt a Lévy flight-based motion model that emulates the irregular and nonlinear movement patterns commonly observed in natural foraging behaviors [9]. Unlike conventional random walks that restrict targets to locally confined regions, Lévy flights generate a mixture of frequent short moves and occasional long jumps, regulated by a heavy-tailed power-law distribution. This biologically inspired

characteristic introduces task complexity, requiring the UAV swarm to adapt to diverse and unpredictable target motions. At each step i, the target position updates as

$$\mathbf{r}_i = \mathbf{r}_{i-1} + s_i \, \hat{\mathbf{d}}_i, \tag{3}$$

where s_i is the step length sampled from a Lévy distribution and $\hat{\mathbf{d}}_i$ is a unit direction vector. To satisfy speed constraints in simulation, steps exceeding the maximum target speed are segmented into smaller moves.

For each simulated trajectory detail, we first generate a set of discrete trajectory points, then connect them into a continuous path, and finally segment the path according to the target's movement step size. where the step length L_i is sampled from a truncated Lévy distribution,

$$L_i = \min\left(\xi^{-\frac{1}{\alpha}}, L_{\max}\right), \qquad \xi \sim \mathcal{U}(0,1), \tag{4}$$

with α controlling the heaviness of the tail and $L_{\max}$ bounding the maximum displacement per points.

The angle evolves in a Markovian manner,

$$\theta_i = (\theta_{i-1} + \Delta\theta) \bmod 2\pi, \qquad \Delta\theta \sim \mathcal{U}(-\pi S, \pi S), \tag{5}$$

where S denotes the turning-strength parameter that governs how abruptly the target can change direction.

Given L_i and θ_i, the updated position is computed as

$$x_i = x_{i-1} + L_i \cos\theta_i, \tag{6}$$

$$y_i = y_{i-1} + L_i \sin\theta_i, \tag{7}$$

each trajectory starts from a random initial location. Excessively long steps are segmented to comply with the maximum speed constraint. Compared with classical random walks, the Lévy flight model produces better reflecting real-world target dynamics and provides a more challenging benchmark for evaluating swarm coordination performance.

3.3. Communication Model

Each UAV maintains a wireless uplink to a GBS to transmit its state information and receive coordinated control commands. We model the communication as a discrete-time process aligned with the control time steps. At each time t, UAV i sends a status packet to the GBS, which may be successfully received or lost depending on network conditions such as channel contention and congestion.

Network congestion and collisions are simulated using the NS-3 environment, incorporating realistic wireless channel modelsprovides the foundation for the traffic control strategy introduced later.

3.4. Baseline Expert Policy: GKA

Within the CSOOC framework, the Greedy Knapsack Algorithm (GKA) serves a dual role: it provides expert supervision for the Teacher Model during the training of the STLF component and acts as an upper-performance reference in subsequent evaluations.

GKA simplifies the CMOMMT task by assuming complete global knowledge of all target positions. At each time step, it executes a two-stage decision process: (1) Point Selection—greedily selecting candidate locations that maximize instantaneous target coverage; and (2) Assignment—employing the Hungarian algorithm to minimize the total UAV travel distance. By combining greedy coverage with optimal assignment, GKA produces near-optimal per-step actions that represent the theoretical upper bound of coordination

performance under ideal network conditions. These expert-generated labels are used to train the Teacher Model in STLF, enabling it to approximate the optimal coordination strategy achievable with full information.

4. Methods

4.1. Overview

The proposed CSOOC framework enables robust UAV swarm coordination under dynamic and unreliable communication conditions by tightly coupling swarm control with real-time network states. Its core objective is to ensure continuous and adaptive multi-target tracking through communication-aware decision-making, CSOOC integrates three fundamental components:

- Online–Offline Coordination Strategy: UAVs dynamically switch between centralized control and local autonomy based on network connectivity. When communication with the GBS is stable, UAVs follow commands from the STLF module (online mode); when congestion or packet loss occurs, they execute a profit-driven local strategy (offline mode) to maintain observation coverage.
- STAR Congestion Control Mechanism: A dual-layer communication regulation approach that combines temporal transmission stratification and RTS/CTS-based channel reservation to reduce packet collisions and stabilize uplink reliability.
- Python–NS-3 Co-Simulation Platform: A socket-based bidirectional coupling between the STLF decision engine and the NS-3 network simulator, enabling synchronized evaluation of control and communication performance within each simulation step.

By unifying decision adaptation, congestion regulation, and cross-domain simulation, CSOOC bridges the gap between algorithmic coordination and real-world network dynamics, offering a framework for cooperative multi-UAV tracking in communication-constrained IoT environments.

4.2. Online and Offline Actions

The proposed CSOOC enables UAV swarms to adapt their decision-making strategy according to real-time communication status. When communication with the GBS is reliable, the swarm operates in online mode, executing globally coordinated commands generated by a centralized learning-based controller. When communication is unstable or fails, UAVs switch to offline mode, relying on local observations and autonomous decision-making. This dual-mode design ensures robust and continuous operation of UAV swarms under both ideal and degraded network conditions.

4.2.1. Online Mode: STLF-Based Centralized Control

When communication with the GBS is successful, UAVs operate in online mode, executing control commands generated by the STLF [9]. The STLF framework learns a near-optimal coordination policy through a Teacher Model trained with privileged global information and then transfers this knowledge to a Student Model adapted to partial observability. As a result, UAVs can perform coordinated decision-making under realistic sensing constraints, maintaining high tracking efficiency.

Two-Stage Learning Paradigm

As illustrated in Figure 4, the STLF consists of two stages:

Stage 1–Teacher Model: trained using expert demonstrations from the GKA, with access to full global information. It learns to output near-optimal movement actions that maximize the overall observation rate.

Stage 2–Student Model: initialized from the Teacher Model and fine-tuned under partial observability. Privileged global data are replaced by an Observable Value Map (OVM), which probabilistically estimates unobserved target distributions.

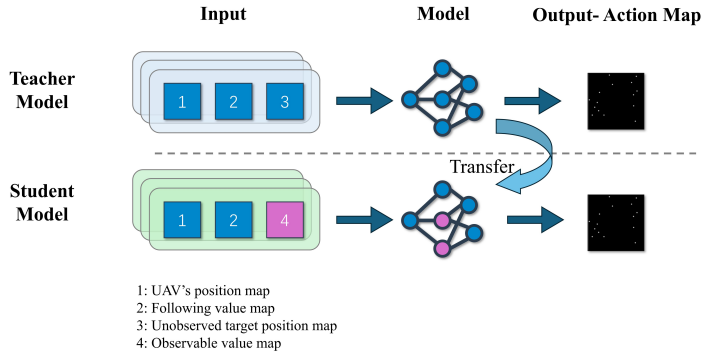


Figure 4. Two-stage STLF framework: Teacher Model training under complete information (Stage 1) and Student Model deployment under partial observability (Stage 2).

Gaussian-Based State Representation

At each time step, the environment is represented by four spatial maps that encode UAV states and target-related information as smooth Gaussian distributions. Unlike binary occupancy maps, Gaussian representations provide continuous and differentiable spatial cues, capturing positional uncertainty and improving the model's spatial reasoning capability.

Specifically, as illustrated in Figure 5, the input to the neural network consists of:

UAV Position Map: encodes the current positions of all UAVs.

Following Value Map: represents targets currently observed within any UAV's FoV, modeled as Gaussian peaks centered on detected target locations.

Unobserved Target Map (Teacher only): provides the privileged global positions of targets outside all UAVs' FoVs during Teacher Model training.

Observable Value Map (Student only): provides a probabilistic estimation of unobserved targets, constructed from historical observations during Student Model deployment.

These four maps are stacked as a multi-channel input tensor and fed into the STLF model, allowing it to jointly consider both tracking and exploration under different information settings.

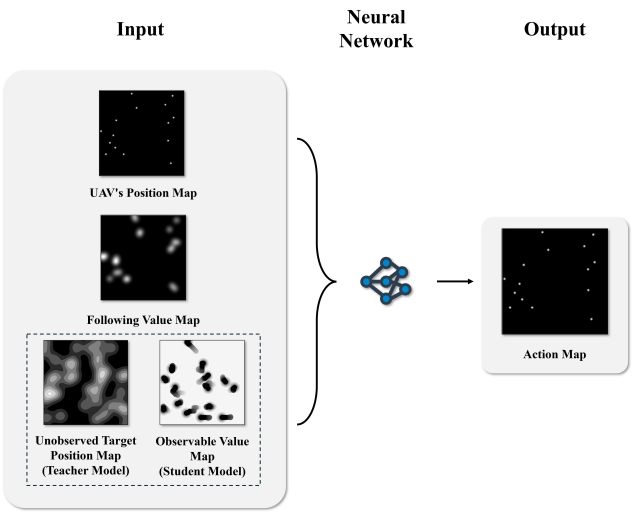


Figure 5. Neural network input–output structure. Four Gaussian-encoded input maps are aggregated and processed by the STLF model to produce an action map for UAV swarm control.

Centralized Action Generation

The stacked input tensor is processed by the residual CNN backbone of the STLF model, which consists of downsampling, residual, and upscaling modules. The output is an action map indicating the next-step movement decision for each UAV. These commands are generated at the GBS, which aggregates state information from all UAVs and broadcasts the resulting control signals back to the swarm. This centralized control mechanism enables globally coordinated behaviors under ideal communication conditions.

4.2.2. Offline Mode: Profit-Driven Local Control

When communication with the GBS fails due to congestion, packet loss, or excessive delay, UAVs automatically switch to offline mode. In this mode, UAVs can no longer receive centralized commands from the STLF model and must make fully autonomous mobility decisions. The primary goal of the offline strategy is to maintain effective target observation and exploration despite the absence of global coordination. This offline mechanism complements the online mode within the CSOOC, ensuring operational continuity under degraded communication conditions.

Local Map Construction

Inspired by the structure of the STLF input representation, each UAV maintains two local, Gaussian-based maps derived entirely from its own historical observations: the Local Following Value Map (F_{local}) and the Local Observable Value Map (O_{local}). These maps serve as lightweight surrogates of the global STLF input, enabling UAVs to make autonomous decisions when communication is unavailable.

Local Following Value Map (F_{local}): This map encodes the locations of targets currently detected within the UAV's sensing range. Each observed target at position (μ_x, μ_y) is represented by an isotropic 2D Gaussian distribution

$$f(x,y) = \frac{1}{2\pi\sigma^2} \exp\left[-\frac{(x-\mu_x)^2 + (y-\mu_y)^2}{2\sigma^2}\right], \tag{8}$$

where σ corresponds to the UAV's observation radius. Multiple target observations are superposed, while the absence of targets results in a zero map, shifting the UAV's behavior toward exploration.

Local Observable Value Map (O_{local}): This map provides a probabilistic estimate of where unobserved targets are most likely to appear, based solely on the UAV's past observations. It integrates a temporal component—capturing increasing uncertainty with time since last visit—and a spatial component—biasing exploration toward regions just beyond the UAV's current FoV. σ_o controls the spatial spread of this exploration bias.

Both maps are updated locally at each time step: F_{local} reflects the latest sensing results, while O_{local} accumulates temporal decay and spatial bias information. This design allows each UAV to maintain a compact yet informative representation of its local environment, forming the foundation for profit-driven offline decision-making. Although the structure is conceptually aligned with the global STLF representation (Figure 5), the local maps only encode state information perceived by a single UAV, without aggregating global observations from the entire swarm.

Profit Function and Movement Selection

At each offline step, the UAV evaluates all candidate positions $p \in P_{\text{candidate}}$ within its mobility range. For each candidate, the expected profit is defined as the total increase in

local map values obtained after moving to p, compared with the map constructed at the UAV's current position p_{current}:

$$\text{Profit}(p) = \sum_{x,y} [M_{\text{local}}(x, y|p) - M_{\text{local}}(x, y|p_{\text{current}})], \tag{9}$$

where M_{local} represents the combined local following and observable value maps.

The UAV then selects the movement that maximizes this profit:

$$p_{\text{next}} = \arg \max_{p \in P_{\text{candidate}}} \text{Profit}(p). \tag{10}$$

This formulation explicitly compares the candidate map with the map generated at the current position, ensuring that the UAV chooses the location yielding the greatest net information gain.

Behavioral Consistency and Advantages

This profit-driven strategy allows UAVs to autonomously maintain effective coverage during communication outages by exploiting local spatiotemporal information. Because F_{local} and O_{local} share the same Gaussian structure as the online STLF inputs, offline actions remain behaviorally consistent with the online mode, enabling smooth policy switching once communication is restored. This ensures that the CSOOC can maintain swarm-level functionality even in the presence of severe communication degradation.

Although CSOOC does not incorporate an explicit collision-avoidance module, it achieves implicit separation through its utility-map design. In the online stage, targets observed by any UAV are removed from the Unobserved Target Map, lowering the utility in that area and naturally preventing UAVs from converging on the same region. During communication loss, the offline policy follows locally constructed profit maps, where previously visited or low-value areas remain unattractive, encouraging further spatial dispersion.

4.3. STAR: Stratified Transmission and RTS/CTS Congestion Control

The proposed coordination framework relies on timely feedback from the GBS. However, a significant challenge arises from the synchronized nature of the UAV swarm: at the beginning of each discrete time step, all UAVs simultaneously attempt to transmit their status updates to the base station. This concurrent transmission leads to severe network congestion, packet collisions, and ultimately communication failures that degrade overall observation performance.

To address this issue, we propose a dual-layer congestion control mechanism termed STAR. STAR integrates temporal desynchronization of UAV transmissions with an RTS/CTS-based channel reservation protocol, providing robust uplink reliability under high-density swarm scenarios.

4.3.1. Stratified Transmission Timing

Instead of having all UAVs transmit at the same instant within a time step, we employ a stratified sampling strategy to distribute transmissions over each one-second interval. Specifically, time is divided into 1-s strata (buckets) denoted as $[k, k+1)$, where $k \in \mathbb{Z}_{\geq 0}$ is the discrete time step.

For each UAV $i \in \{1, 2, \ldots, N\}$, the transmission time within the k-th stratum is determined by

$$t_i^{(k)} = k + U_{i,k}, \quad U_{i,k} \sim \mathcal{U}(0, 1), \tag{11}$$

where $U_{i,k}$ is an independent random variable that follows a uniform distribution over the interval $[0, 1]$. This means that all values in $[0, 1]$ are equally likely to be selected, introducing randomness without bias toward any specific moment within the stratum.

This formulation ensures that each UAV transmits exactly one packet per interval, while the random offset $U_{i,k}$ introduces temporal diversity among UAVs. By avoiding phase-locking of transmission schedules, the probability of repeated collisions is reduced. In practice, each UAV maintains an independent timer that triggers the packet transmission at the sampled time $t_i^{(k)}$ within each stratum.

4.3.2. RTS/CTS-Enhanced Handshake

To further reduce packet collisions during the uplink phase, STAR incorporates a Request-to-Send/Clear-to-Send (RTS/CTS [14]) mechanism into the communication protocol. Before transmitting its data packet, each UAV first sends a short RTS frame to the base station. The base station, upon receiving an RTS, responds with a CTS frame if the channel is clear, thereby reserving the channel for the corresponding UAV.

The handshake success of UAV i at transmission attempt t_i can be modeled as

$$S_i = \begin{cases} 1, & \text{if UAV } i \text{ receives a CTS reply from the GBS,} \\ 0, & \text{otherwise,} \end{cases} \tag{12}$$

where S_i is a binary variable indicating whether the RTS/CTS exchange is successful.

This handshake mechanism mitigates hidden terminal problems and minimizes the chance of data packet collisions, which are more costly due to their larger size. The combination of temporally staggered transmissions and channel reservation significantly improves the reliability and throughput of the uplink channel. The STAR scheme is implemented within the NS-3 simulation environment, where the MAC-layer behavior explicitly models the timing and handshake mechanisms.

In addition to reducing packet collisions, STAR addresses bandwidth limitations and latency fluctuations in dense UAV swarms. By assigning stratified transmission windows, STAR prevents a large number of UAVs from contending for the channel simultaneously, which reduces instantaneous congestion.

4.4. NS-3 with Python's Communication

To simulate the impact of network conditions on swarm coordination, we establish a bidirectional communication link between the Python-based CMOMMT environment simulation and the NS-3 network simulator. This integration is achieved through a socket-based interface, enabling real-time data exchange between the two simulation environments. The communication process is designed to operate in a discrete-time manner, with each time step t corresponding to a synchronized simulation cycle in both platforms. Figure 6 shows the interaction description at each discrete moment.

At the beginning of each time step, the Python environment sends the current positions to the NS-3 simulator via a TCP socket. This data packet includes the coordinates of each UAV, which are used by NS-3 to model wireless transmission between the UAVs and the GBS. The NS-3 simulator then emulates the uplink communication process, incorporating factors such as signal propagation, interference, packet collisions, and queuing delays based on the configured network model. The output of this simulation—namely, the success or failure of each UAV's transmission—is returned to the Python side through the same socket connection.

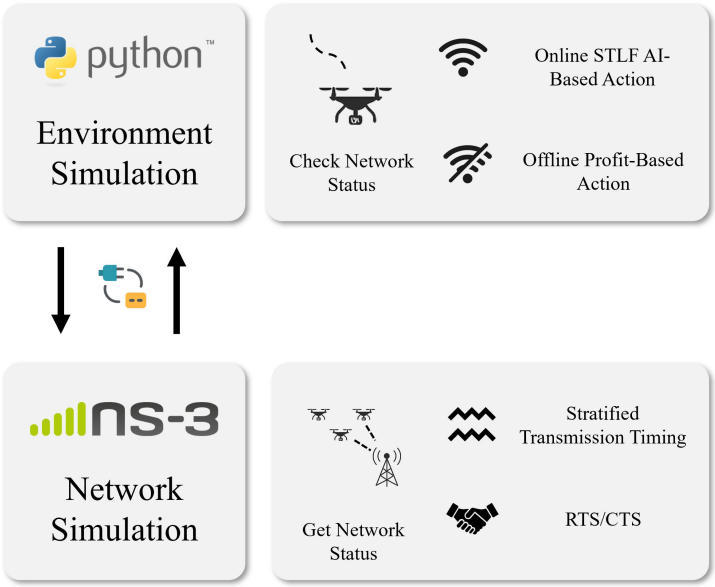


Figure 6. Conceptual overview of the proposed CSOOC framework, integrating STLF-based centralized control, profit-driven offline autonomy, and STAR congestion control within a unified communication–control architecture.

On the Python side, the CMOMMT simulation environment processes the received communication feedback to determine the operational mode for each UAV. If a UAV successfully receives a control command from the GBS within the current time step, it operates in online mode, executing the action generated by the STLF model. If communication fails—due to packet loss, congestion, or delay—the UAV switches to offline mode, where it follows a predefined mobility strategy based on a local Gaussian profit map. This map integrates both exploration and tracking incentives, allowing the UAV to continue mission-critical behavior without central guidance.

The socket interface ensures that both simulators remain synchronized throughout the mission duration. Each simulation step in Python corresponds to a fixed time interval in NS-3, preserving temporal consistency between the mobility and network layers. This co-simulation framework allows us to evaluate the robustness of the online policy under varying network conditions, providing a more realistic assessment of swarm performance in communication-constrained environments.

5. Experiment

5.1. Experimental Setup

Experiments were implemented in Python 3.9 using the PyTorch 2.9.1 deep learning framework and trained on an NVIDIA V100 GPU. The network simulation component was deployed on a VMware® Workstation 16 Pro virtual machine (version 16.2.3) running Ubuntu 22.04.2 LTS. The NS-3 network simulator (version 3.44) was used to emulate wireless communication, including congestion, packet loss, and latency effects, through its built-in Wi-Fi and LTE modules. The Python-based STLF environment and NS-3 were synchronized via TCP sockets to ensure consistent discrete-time communication and control cycles throughout each simulation episode.

Table 1 outlines the simulation parameters used to instantiate the CMOMMT scenario. The search area is defined as a 400×400 grid containing 60 targets. UAV swarm sizes are configured as 15, 20, and 30, corresponding to UAV-to-target ratios of 1:4, 1:3, and 1:2, respectively. Observation radii are specified as 12 (small), 18 (medium), and 24 (large) cells. UAVs and targets move at maximum speeds of 3 and 1 cells per time step, respectively. Each

simulation runs for 500 time steps. All results are averaged over five independent trials to reduce stochastic variance introduced by random initialization and transmission timing.

In the NS-3 configuration, UAVs communicate with the ground base station over Wi-Fi (IEEE 802.11n) with a fixed transmission power of 10 dBm, with 20 MHz. Each UAV transmits one UDP packet of size 1024 Bytes per time step, enabling accurate modeling of congestion and packet loss under high-density scenarios.

For Gaussian smoothing used in input maps, the standard deviation σ is defined based on the input type: in the UAV's position map, σ is set to the UAV's maximum movement range per time step; in the following value map, σ equals the UAV's observation radius; and in the unobserved target position map, σ is set to 1.5 times the UAV's observation radius. This design ensures that each input map accurately represents its respective spatial influence and observation uncertainty.

Table 1. Simulation Parameters.

Parameter	Value
Search Region	400×400 Cells
Number of Targets	60
Number of UAVs	15/20/30
UAV Observation Radius	12/18/24 Cells
UAV Maximum Speed	3 Cells per Step
Target Maximum Speed	1 Cell per Step
Mission Execution Period	500 Time Steps

5.2. Evaluation Metrics

To quantitatively evaluate the performance of CSOOC, we adopt two categories of metrics:

(1) Task-level metrics (CMOMMT performance): The primary objective metric is the Average Observation Rate (AOR), defined as the ratio between the total number of targets observed at each time step and the total number of targets in the environment. A higher AOR reflects better overall swarm coverage and tracking efficiency across the mission.

(2) Network-level metrics (communication quality): To characterize the impact of network conditions on swarm coordination, we measure:

Packet Delivery Ratio (PDR): the percentage of successfully received packets at the GBS over the total transmitted packets.

Latency: the average end-to-end transmission delay for packets sent from UAVs to the GBS.

Jitter: the variation in packet arrival time, indicating the stability of the wireless link.

These metrics jointly capture both the mission performance and the underlying network reliability, enabling a comprehensive evaluation of the proposed framework under different communication conditions.

5.3. Validation with NS-3 Network Simulation

To illustrate the performance degradation caused by communication impairments, a relatively small-scale configuration is adopted, consisting of 15 UAVs with an observation radius of 12 units. Figure 7 illustrates the average observation rate curves under this setting. For completeness, Figure 7 also visualizes the standard deviation of each curve as a shaded region. The SD values are 0.76% for GKA with NS-3, 0.78% for GKA without NS-3, 1.56% for STLF with NS-3, and 1.16% for STLF without NS-3.

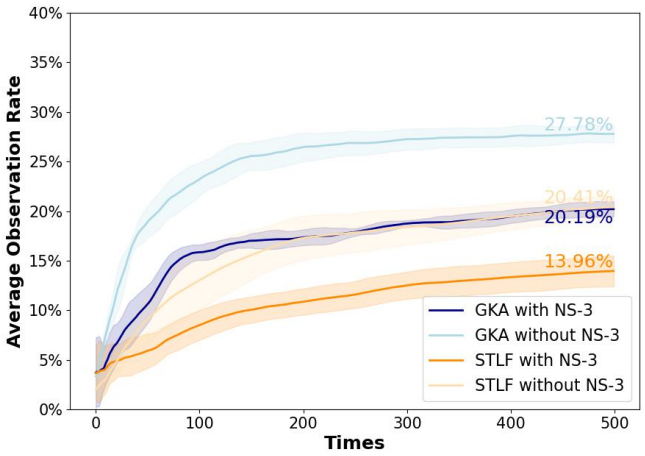


Figure 7. Average observation rate under ideal and NS-3 simulated network conditions.

In the lines with NS-3, both the STLF and the GKA exhibit stable tracking performance, yet a clear performance gap remains. GKA, benefiting from complete global information, consistently generates near-optimal assignments with a higher observation rate. In contrast, STLF, although able to approximate the expert strategy through knowledge transfer, achieves slightly lower performance due to partial observability constraints.

When the NS-3 network simulation is integrated into the mission workflow, the overall performance declines. Since all UAVs transmit their status simultaneously at each discrete time step, network congestion, packet collisions, and additional delays occur. These impairments particularly affect those two methods, where unstable communication leads to reduced accuracy of coordinated decisions, thereby lowering the average observation rate compared to the ideal case. This experiment also verifies that our socket-based co-simulation mechanism successfully bridges the Python-based CMOMMT environment and the NS-3 network simulation. The observed degradation in both network quality and observation performance is fully consistent with expectations, confirming the correctness and effectiveness of the integration.

Table 2 reports the network statistics. It can be observed that the PDR is below 100% for both methods, with values of 75.35% for GKA and 71.91% for STLF. Meanwhile, the average latency remains around 5 ms, and the jitter is measured at 3.26 ms and 3.61 ms, respectively. These results suggest that although the network is still operational, packet loss and jitter have a direct negative impact on centralized coordination, thus reducing overall tracking performance.

Table 2. NS-3 network metrics for GKA and STLF.

Method	GKA	STLF
PDR (%)	75.35	71.91
Latency (ms)	5.33	5.38
Jitter (ms)	3.26	3.61

In summary, Figure 7 and Table 2 jointly demonstrate that: (1) under ideal communication, GKA consistently outperforms STLF; (2) once realistic network effects are introduced, both algorithms experience performance degradation; and (3) the proposed Python–NS-3 socket-based co-simulation framework accurately reflects network-induced performance loss, validating its effectiveness for communication-aware UAV swarm coordination studies.

To wider validation of NS-3, we examine the impact of different channel bandwidths, we evaluate both methods under 10 MHz and 40 MHz settings, while keeping 20 MHz as the default configuration. The 20 MHz bandwidth is consistent with the standard IEEE 802.11 PHY configuration used throughout our NS-3 simulations, and it provides sufficient capacity for the experimental setup. As shown in Table 3, performance remains similar when the bandwidth is increased to 40 MHz, confirming that the default setting is already adequate for the given traffic load. In contrast, reducing the bandwidth to 10 MHz leads to noticeable degradation in both PDR and AOR, indicating that the network becomes congested when channel capacity is limited.

Table 3. Performance under different bandwidth settings.

Metrics	10 MHz	20 MHz	40 MHz
GKA			
AOR (%)	17.83	20.19	20.03
PDR (%)	63.87	75.35	74.40
Latency (ms)	14.15	5.33	1.14
Jitter (ms)	11.03	3.26	3.42
STLF			
AOR (%)	10.73	13.96	14.21
PDR (%)	63.90	71.91	73.33
Latency (ms)	13.00	5.38	3.62
Jitter (ms)	10.13	3.61	2.82

5.4. Validation of STAR Congestion Control

To evaluate the effectiveness of the proposed STAR mechanism, we compare network performance and swarm tracking results with and without STAR under identical communication and mission parameters. Specifically, we conduct experiments using the 15 UAVs with an observation radius of 12 units setting and evaluate both communication metrics and swarm-level observation performance.

The results in Table 4 clearly demonstrate that introducing the STAR congestion control mechanism significantly improves network reliability and swarm coordination performance. Specifically, the PDR increases from 75.35% to 92.96% for GKA and from 71.91% to 94.22% for STLF, indicating a substantial reduction in packet collisions and transmission failures. As network stability improves, the UAV swarm can receive more accurate coordination commands, leading to more effective target coverage and tracking.

Table 4. Performance comparison with and without STAR congestion control.

Metrics	With STAR	Without STAR
GKA		
AOR (%)	22.99	20.19
PDR (%)	92.96	75.35
Latency (ms)	5.82	5.33
Jitter (ms)	1.88	3.26
STLF		
AOR (%)	17.44	13.96
PDR (%)	94.22	71.91
Latency (ms)	5.85	5.38
Jitter (ms)	3.52	3.61

Moreover, STAR stabilizes the network by effectively reducing jitter. For example, the jitter for GKA decreases from 3.26 ms to 1.88 ms, suggesting more predictable and

less bursty transmission delays. Although the average latency slightly increases for both algorithms (e.g., from 5.33 ms to 5.82 ms for GKA), this is a reasonable trade-off considering the significant gains in delivery reliability and stability. The additional delay stems from the temporal stratification component of STAR, which intentionally staggers UAV transmissions to reduce collisions.

Overall, these findings validate the effectiveness of the proposed STAR congestion control mechanism. By jointly applying temporal desynchronization and channel reservation, STAR enhances network stability and improves swarm coordination performance under dense communication scenarios.

5.5. Validation of Offline Action Within CSOOC

To isolate and evaluate the contribution of offline action within the proposed CSOOC, we conduct controlled experiments under communication uncertainty. These experiments are performed using the same configuration as the network validation experiment—15 UAVs with an observation radius of 12 units.

When communication feedback from the GBS is lost, UAVs operating under CSOOC transition from online mode to the offline action module, which is driven by Gaussian-based local profit maps. Rather than stalling or performing random movements, each UAV makes autonomous mobility decisions based on its own historical observations, dynamically balancing between target following and exploratory search. This design provides behavioral continuity in the absence of centralized guidance, maintaining effective coverage over moving targets.

As shown in Table 5, introducing offline actions leads to a clear improvement in tracking performance. For the GKA baseline, the AOR increases from 20.19% to 23.46%, while for STLF, it increases from 13.96% to 17.78%. The PDR also improves, reflecting the more adaptive behavior of UAVs during periods of unstable communication. Figure 8 further illustrates how the average observation rate evolves over time, showing improvements for both GKA and STLF when offline action is enabled.

These results highlight the critical role of the offline action module in CSOOC:

- It preserves coordinated swarm behavior through local decision-making, even when global control is temporarily unavailable.
- It ensures graceful performance degradation instead of abrupt mission failure during communication disruptions.
- It provides a bridge between online and offline modes, maintaining strategy consistency with the STLF policy.

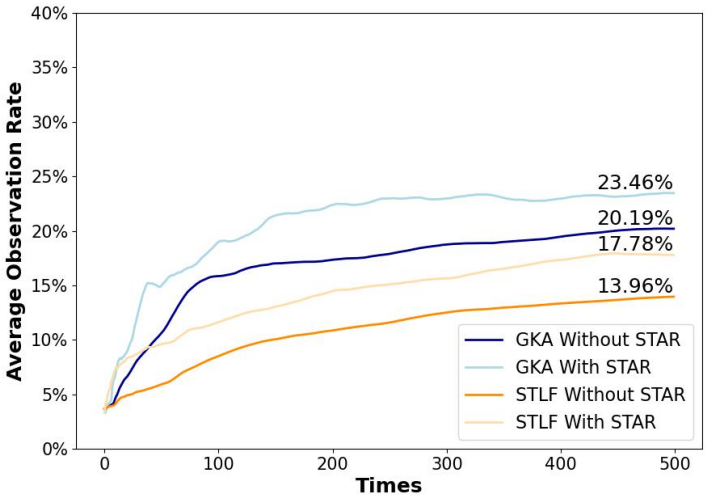


Figure 8. Average observation rate evolution with and without STAR.

Table 5. Performance comparison with and without offline action.

Metrics	With Offline Action	Without Offline Action
GKA		
AOR (%)	23.46	20.19
PDR (%)	82.55	75.35
Latency (ms)	6.36	5.33
Jitter (ms)	3.96	3.26
STLF		
AOR (%)	17.78	13.96
PDR (%)	80.96	71.91
Latency (ms)	6.35	5.38
Jitter (ms)	3.99	3.61

In summary, the offline module acts as an essential behavioral backbone of the CSOOC framework, enabling UAV swarms to maintain robust observation performance under communication uncertainty without requiring external intervention.

5.6. Validation of the CSOOC and Comparative Analysis

To comprehensively validate the effectiveness of the proposed CSOOC framework, its performance is compared with several baseline algorithms under varying UAV maximum speeds and UAV-to-target ratios. In previous experiments, the contributions of the offline action mechanism and the STAR congestion control were individually examined; here, both components are jointly incorporated into all comparative methods to ensure fairness and isolate the effect of coordination strategy. To further assess the generalization capability of CSOOC, evaluations are conducted across different swarm scales (15, 20, and 30 UAVs) and observation radii (12, 18, and 24 cells), which effectively vary the UAV-to-target ratio and spatial coverage conditions within the CMOMMT environment. The benchmarking algorithms include:

- GKA: A near-optimal heuristic that serves as the performance upper bound.
- CSOOC: An enhanced version of STLF that incorporates a profit-driven mobility strategy and congestion control for communication-aware swarm coordination.
- PAMTS [5]: PAMTS is a profit-driven search-and-follow algorithm that coordinates UAVs using an observation profit metric, guiding each UAV at every time step toward the location with the highest expected profit.
- PAMTS Follow [5]: A variant of PAMTS that prioritizes target following upon detection. In contrast to the original PAMTS, which dynamically balances exploration and tracking, PAMTS Follow disables exploration by setting its weight to zero, improving observation consistency, especially in tasks requiring persistent surveillance, such as convoy protection or long-term monitoring.
- A-CMOMMT [4]: The earliest proposed solution to the CMOMMT task.
- I-CMOMMT [7]: Built upon A-CMOMMT, I-CMOMMT incorporates an idleness vector to account for the time since a region was last observed.
- ROD (Reward-based Orientation Determination) [8]: The ROD method is included for comparison, as it determines UAV movement directions based on reward evaluation to optimize cooperative target search efficiency.
- VDP-ACO (Voronoi-based Dynamic Partition with Ant Colony Optimization) [33]: A distributed framework that integrates Voronoi-based dynamic area partitioning and ant colony optimization under the distributed model predictive control architecture to enhance cooperative target search efficiency in dynamic environments.

The baselines selected in this study are representative used in CMOMMT research, covering classical, optimization-based, and learning-enhanced coordination strategies that directly address the characteristics of the multi-target tracking task. While DRL coordination algorithms are also of interest, our focus here is to evaluate CSOOC against task-specific and domain-relevant approaches. Incorporating DRL-based coordination methods constitutes a meaningful extension for future work.

As shown in Figure 9 and Table 6, the average observation rate of all methods increases consistently with both the number of UAVs and the observation radius. This trend aligns with expectations, as larger swarms and wider sensing ranges allow broader spatial coverage and more frequent target detections. Across all configurations, the gap between methods gradually narrows with increasing swarm size, reflecting the diminishing marginal benefit of coordination when coverage becomes saturated. Nevertheless, distinct performance hierarchies remain evident across methods.

The proposed CSOOC demonstrates stable and superior performance across all tested configurations. By integrating a profit-driven offline module and the STAR congestion control mechanism, CSOOC effectively maintains high observation rates under both sparse and dense swarm conditions. Compared to the original STLF, CSOOC exhibits stronger adaptability to varying UAV–target ratios, sustaining coordinated behavior even when communication intermittency occurs. Its consistent improvement across all radii highlights the framework's capability to seamlessly couple centralized and decentralized decision-making under communication uncertainty.

In contrast, traditional heuristic approaches show limited robustness. PAMTS and A-CMOMMT depend heavily on parameter tuning and pre-defined heuristics, which restrict their adaptability when the environment scale changes. PAMTS Follow and ROD display strong behavioral biases—favoring tracking and exploration respectively—resulting in poor balance between persistent observation and spatial exploration. I-CMOMMT achieves moderate improvements by incorporating idleness awareness but remains constrained by the lack of communication feedback modeling. VDP-ACO exhibits fluctuating results, performing competitively in medium-scale cases but losing effectiveness in dense scenarios due to its distributed optimization overhead.

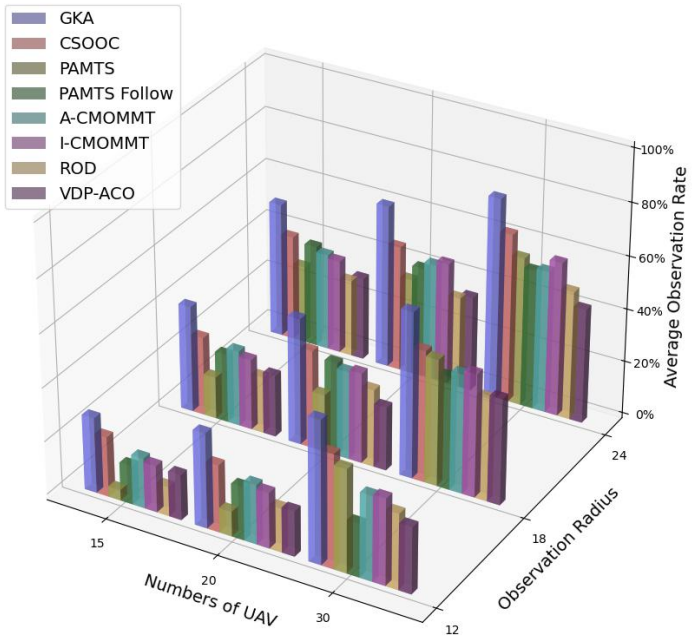


Figure 9. Average observation rate comparison under different UAV counts and observation radii.

Table 6. Average Observation Rate (%) for different UAV numbers and observation radii.

r	Algorithm	15 UAV	20 UAV	30 UAV
12	GKA	28.23 ± 1.21	35.50 ± 2.11	53.47 ± 2.84
	CSOOC	22.47 ± 0.73	25.15 ± 1.42	42.57 ± 2.05
	PAMTS	3.93 ± 0.02	9.26 ± 0.13	38.58 ± 1.85
	PAMTS Follow	14.50 ± 1.39	19.95 ± 2.23	36.99 ± 1.96
	A-CMOMMT	18.83 ± 0.91	22.24 ± 1.64	31.69 ± 1.47
	I-CMOMMT	17.42 ± 0.66	20.80 ± 1.15	32.32 ± 1.99
	ROD	10.98 ± 0.31	16.48 ± 0.53	28.29 ± 1.58
	VDP-ACO	16.98 ± 1.57	19.96 ± 0.86	24.82 ± 1.39
18	GKA	40.15 ± 1.27	47.70 ± 1.93	62.40 ± 2.88
	CSOOC	29.63 ± 0.74	36.91 ± 1.12	49.42 ± 2.41
	PAMTS	15.73 ± 1.03	21.64 ± 2.08	47.56 ± 3.08
	PAMTS Follow	28.50 ± 1.44	35.08 ± 1.34	42.34 ± 3.19
	A-CMOMMT	28.58 ± 0.96	33.16 ± 2.31	44.82 ± 2.57
	I-CMOMMT	26.62 ± 1.51	34.08 ± 1.27	46.38 ± 2.02
	ROD	21.38 ± 0.85	29.12 ± 2.46	38.69 ± 2.91
	VDP-ACO	23.36 ± 1.63	23.96 ± 1.18	39.95 ± 1.99
24	GKA	51.17 ± 2.21	61.84 ± 2.16	75.99 ± 3.12
	CSOOC	39.84 ± 1.08	47.50 ± 1.57	63.87 ± 2.65
	PAMTS	29.55 ± 1.93	36.21 ± 2.44	56.24 ± 1.88
	PAMTS Follow	38.61 ± 1.36	41.80 ± 1.73	52.68 ± 2.07
	A-CMOMMT	36.71 ± 1.77	44.44 ± 2.03	53.76 ± 2.49
	I-CMOMMT	35.60 ± 1.12	46.22 ± 1.82	58.29 ± 3.31
	ROD	29.16 ± 1.25	34.35 ± 1.66	48.70 ± 1.93

Quantitatively, CSOOC achieves an average observation rate of 39.7% across all parameter configurations, surpassing other baseline algorithms by 4.4–11.13%. These results confirm that CSOOC not only sustains stable coordination under communication constraints but also scales efficiently across different swarm sizes and sensing ranges, establishing it as a robust and generalizable solution for real-world CMOMMT tasks.

The network performance metrics summarized in Table 7 further validate the effectiveness of the STAR congestion control mechanism integrated into all methods. Compared with the corresponding results in Table 4 (without STAR), every algorithm exhibits noticeable improvements in packet delivery ratio (PDR), latency, and jitter, confirming that STAR effectively mitigates transmission contention and stabilizes network behavior. The average PDR values increase by 13.68–21.48% across all configurations, while both latency and jitter decrease, reflecting more reliable and smoother uplink communication between UAVs and the ground base station. To avoid enumerating all entries of Table 7, we highlight the key trends that summarize the overall network behavior. The STAR mechanism consistently maintains a high packet delivery ratio, typically above 85–90% across different swarm sizes and observation radii. This indicates that STAR effectively mitigates congestion and improves communication stability in different UAV numbers. CSOOC keeps the end-to-end latency below 10 ms in all tested configurations, demonstrating that the framework preserves real-time responsiveness under varying network loads.

It is noteworthy that the GKA and CSOOC do not always achieve the absolute best network metrics among all tested methods. This variation, however, is not a limitation of the proposed framework but rather a result of spatial topology factors. Since the ground base station is located at the center of the environment (200, 200), UAVs positioned closer to the center experience more stable connections, while those distributed toward the map boundaries suffer from weaker signal strength and higher contention levels. Therefore,

the overall network statistics are influenced by the spatial dispersion of UAVs and their instantaneous proximity to the base station rather than by the control algorithm itself.

Despite these variations, CSOOC maintains a consistently high PDR (above 90%) and low latency (below 10 ms) across all swarm configurations. This stability ensures timely feedback for the online mode and reliable switching between online and offline behaviors during communication fluctuations. The results collectively demonstrate that STAR effectively enhances the underlying communication layer and that CSOOC can preserve robust coordination performance even when physical topology or link quality varies dynamically.

Table 7. Network performance under different UAV numbers and observation radii. PDR in %, Delay/Jitter in ms.

		15 UAV			**20 UAV**			**30 UAV**		
r	**Algorithm**	**PDR**	**Delay**	**Jitter**	**PDR**	**Delay**	**Jitter**	**PDR**	**Delay**	**Jitter**
12	GKA	92.97	5.40	1.57	89.47	11.41	8.24	91.06	5.37	1.63
	CSOOC	93.36	6.19	2.57	88.93	10.71	7.48	86.55	6.04	2.63
	PAMTS	92.18	10.66	8.66	87.30	13.3	11.01	90.10	10.06	7.10
	PAMTS Follow	94.01	5.03	1.41	90.01	9.96	7.24	88.10	8.82	5.77
	A-CMOMMT	93.10	5.56	1.78	88.69	11.04	7.71	91.24	5.82	2.17
	I-CMOMMT	91.74	6.10	2.57	91.98	10.13	6.46	87.66	9.57	6.07
	ROD	91.14	6.49	3.12	87.89	10.68	7.05	86.58	11.27	9.37
	VDP-ACO	92.86	6.55	3.62	91.79	11.55	8.38	88.07	7.98	5.37
18	GKA	91.38	5.65	2.06	91.51	5.67	1.88	90.49	5.64	1.87
	CSOOC	90.34	6.81	3.26	90.06	6.13	2.76	87.83	11.10	7.55
	PAMTS	90.37	10.27	7.93	87.07	17.92	15.32	91.05	14.49	11.65
	PAMTS Follow	93.48	8.09	5.21	86.23	14.64	11.35	93.19	8.88	5.84
	A-CMOMMT	91.76	11.59	8.11	90.32	14.73	11.01	90.08	12.07	9.88
	I-CMOMMT	91.71	9.22	6.28	89.37	16.85	13.63	90.67	12.94	9.51
	ROD	85.17	10.76	8.42	89.25	12.89	9.96	85.30	15.46	12.41
	VDP-ACO	90.24	9.11	7.05	90.27	16.95	13.52	87.68	12.59	10.34
24	GKA	92.93	5.82	1.92	92.00	5.90	1.72	88.45	5.33	1.74
	CSOOC	90.80	6.80	3.11	91.03	8.85	5.43	89.38	9.31	5.57
	PAMTS	89.28	10.53	8.14	88.76	7.84	4.53	91.57	7.67	4.77
	PAMTS Follow	93.39	6.52	2.51	89.38	7.15	3.82	87.06	7.63	4.89
	A-CMOMMT	91.47	7.32	3.55	91.89	8.53	5.90	86.29	8.83	6.31
	I-CMOMMT	92.29	7.78	4.67	90.48	7.97	4.78	86.26	15.46	11.03
	ROD	91.36	7.68	3.92	91.75	7.45	4.33	87.35	10.58	7.06
	VDP-ACO	91.83	7.34	4.21	86.97	8.52	5.97	85.59	10.76	7.97

6. Conclusions and Future Work

This paper presented CSOOC, a communication-state driven online–offline coordination framework for UAV swarm multi-target tracking under realistic network conditions. Building upon the foundation of the STLF architecture, CSOOC introduces two critical extensions: a profit-driven offline action module for autonomous control during communication outages, and the STAR congestion control mechanism for maintaining uplink stability in dense swarm environments. By dynamically coupling swarm decision-making with network communication states, CSOOC achieves robust, adaptive coordination capable of sustaining high observation performance even under imperfect connectivity.

Extensive experiments conducted through a Python–NS-3 co-simulation platform demonstrated the framework's effectiveness and generalization capability. Results show that CSOOC consistently outperforms baseline algorithms across diverse configurations of UAV counts, observation radii, and target densities. The integration of STAR improves

packet delivery ratios by more than 15% on average, while the offline module ensures behavioral continuity during transient link failures. These findings collectively verify that communication-aware coordination and adaptive control can bridge the gap between algorithmic efficiency and practical UAV swarm deployment in real-world IoT sensing environments. Beyond simulation, CSOOC is well-suited for deployment in practical scenarios where communication quality varies over time. Examples include large-scale agricultural monitoring, disaster response operations, persistent environmental surveillance, and long-range infrastructure inspection, where uplink congestion and intermittent connectivity frequently occur. The STAR mechanism can stabilize packet delivery under these conditions, while the online–offline switching logic enables the swarm to maintain coordinated behavior even during temporary communication loss. Future work will focus on extending the CSOOC framework to decentralized communication architectures and heterogeneous UAV platforms, as well as integrating learning-based congestion prediction and adaptive bandwidth allocation to further enhance scalability and robustness in large-scale aerial networks.

Author Contributions: Conceptualization, H.S. and Y.Y.; methodology and supervision, X.L.; formal analysis, G.L.; investigation, Y.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Science and Technology Development Fund of Macau grant number 0079/2019/AMJ.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Dataset available on request from the authors.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Wei, Z.; Zhu, M.; Zhang, N.; Wang, L.; Zou, Y.; Meng, Z.; Wu, H.; Feng, Z. UAV-assisted data collection for Internet of Things: A survey. *IEEE Internet Things J.* **2022**, *9*, 15460–15483. [CrossRef]
2. Messaoudi, K.; Oubbati, O.S.; Rachedi, A.; Lakas, A.; Bendouma, T.; Chaib, N. A survey of UAV-based data collection: Challenges, solutions and future perspectives. *J. Netw. Comput. Appl.* **2023**, *216*, 103670. [CrossRef]
3. Gupta, L.; Jain, R.; Vaszkun, G. Survey of important issues in UAV communication networks. *IEEE Commun. Surv. Tutor.* **2015**, *18*, 1123–1152. [CrossRef]
4. Parker, L.E. Distributed algorithms for multi-robot observation of multiple moving targets. *Auton. Robot.* **2002**, *12*, 231–255. [CrossRef]
5. Li, X.; Chen, J.; Deng, F.; Li, H. Profit-driven adaptive moving targets search with UAV swarms. *Sensors* **2019**, *19*, 1545. [CrossRef] [PubMed]
6. Parker, L.E.; Emmons, B.A. Cooperative multi-robot observation of multiple moving targets. In Proceedings of the International Conference on Robotics and Automation, Albuquerque, NM, USA, 20–25 April 1997; Volume 3, pp. 2082–2089.
7. Chahal, J.; Belbachir, A.; Seghrouchni, A.E.F. I-CMOMMT: A multiagent approach for patrolling and observation of mobile targets with a continuous environment representation (S). In Proceedings of the 33rd International Conference on Software Engineering & Knowledge Engineering, Pittsburgh, PA, USA, 1–10 July 2021; pp. 21–24.
8. Luo, Q.; Luan, T.H.; Shi, W.; Fan, P. Edge computing enabled energy-efficient multi-UAV cooperative target search. *IEEE Trans. Veh. Technol.* **2023**, *72*, 7757–7771. [CrossRef]
9. Sun, H.; Li, X.; Yan, Y.; Jiang, T.; Liu, B. A Supervised and Transfer Learning Based Two-Stage Framework for UAV Swarm Multi-Target Tracking. *J. Latex Cl. Files* **2025**, *14*. [CrossRef]
10. Nawaz, H.; Ali, H.M.; Laghari, A.A. UAV communication networks issues: A review. *Arch. Comput. Methods Eng.* **2021**, *28*, 1349–1369. [CrossRef]

11. Meng, K.; Wu, Q.; Xu, J.; Chen, W.; Feng, Z.; Schober, R.; Swindlehurst, A.L. UAV-enabled integrated sensing and communication: Opportunities and challenges. *IEEE Wirel. Commun.* **2023**, *31*, 97–104. [CrossRef]
12. Pourjabar, M.; AlKatheeri, A.; Rusci, M.; Barcis, A.; Niculescu, V.; Ferrante, E.; Palossi, D.; Benini, L. Land & localize: An infrastructure-free and scalable nano-drones swarm with UWB-based localization. In Proceedings of the 2023 19th International Conference on Distributed Computing in Smart Systems and the Internet of Things (DCOSS-IoT), Pafos, Cyprus, 19–21 June 2023; pp. 654–660.
13. Nowakowski, M.; Idzkowski, A. Ultra-wideband signal transmission according to European regulations and typical pulses. In Proceedings of the 2020 International Conference Mechatronic Systems and Materials (MSM), Bialystok, Poland, 1–3 July 2020; pp. 1–4.
14. Xu, K.; Gerla, M.; Bae, S. Effectiveness of RTS/CTS handshake in IEEE 802.11 based ad hoc networks. *Ad Hoc Netw.* **2003**, *1*, 107–123. [CrossRef]
15. Kolling, A.; Carpin, S. Cooperative observation of multiple moving targets: An algorithm and its formalization. *Int. J. Robot. Res.* **2007**, *26*, 935–953. [CrossRef]
16. Ding, Y.; Zhu, M.; He, Y.; Jiang, J. P-CMOMMT algorithm for the cooperative multi-robot observation of multiple moving targets. In Proceedings of the 2006 6th World Congress on Intelligent Control and Automation, Dalian, China, 21–23 June 2006; Volume 2, pp. 9267–9271.
17. Ding, Y.; He, Y. Flexible formation of the multi-robot system and its application on cmommt problem. In Proceedings of the 2010 2nd International Asia Conference on Informatics in Control, Automation and Robotics (CAR 2010), Wuhan, China, 6–7 March 2010; Volume 1, pp. 377–382.
18. Li, B.; Wang, J.; Song, C.; Yang, Z.; Wan, K.; Zhang, Q. Multi-UAV roundup strategy method based on deep reinforcement learning CEL-MADDPG algorithm. *Expert Syst. Appl.* **2024**, *245*, 123018. [CrossRef]
19. Yan, J.; Zhang, L.; Yang, X.; Chen, C.; Guan, X. Communication-aware motion planning of AUV in obstacle-dense environment: A binocular vision-based deep learning method. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 14927–14943. [CrossRef]
20. Psomiadis, E.; Maity, D.; Tsiotras, P. Communication-aware map compression for online path-planning. In Proceedings of the 2024 IEEE International Conference on Robotics and Automation (ICRA), Yokohama, Japan, 13–17 May 2024; pp. 12368–12374.
21. Islam, N.; Rashid, M.M.; Pasandideh, F.; Ray, B.; Moore, S.; Kadel, R. A review of applications and communication technologies for internet of things (Iot) and unmanned aerial vehicle (uav) based sustainable smart farming. *Sustainability* **2021**, *13*, 1821. [CrossRef]
22. Eltahlawy, A.M.; Aslan, H.K.; Abdallah, E.G.; Elsayed, M.S.; Jurcut, A.D.; Azer, M.A. A survey on parameters affecting MANET performance. *Electronics* **2023**, *12*, 1956. [CrossRef]
23. ns-3 Consortium. ns-3: Discrete-Event Network Simulator. Available online: https://www.nsnam.org/ (accessed on 22 October 2025).
24. Piro, G.; Baldo, N.; Miozzo, M. An LTE module for the ns-3 network simulator. In Proceedings of the SimuTools, Barcelona, Spain, 22–24 March 2011; pp. 415–422.
25. Gill, J.S.; Velashani, M.S.; Wolf, J.; Kenney, J.; Manesh, M.R.; Kaabouch, N. Simulation testbeds and frameworks for UAV performance evaluation. In Proceedings of the 2021 IEEE International Conference on Electro Information Technology (EIT), Mt. Pleasant, MI, USA, 14–15 May 2021; pp. 335–341.
26. Jiang, W.; Han, H.; He, M.; Gu, W. Network simulation tools for unmanned aerial vehicle communications: A survey. *Int. J. Commun. Syst.* **2024**, *37*, e5878. [CrossRef]
27. Baidya, S.; Shaikh, Z.; Levorato, M. Flynetsim: An open source synchronized uav network simulator based on ns-3 and ardupilot. In Proceedings of the 21st ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, Montreal, QC, Canada, 28 October–2 November 2018; pp. 37–45.
28. Garg, S.; Ihler, A.; Bentley, E.S.; Kumar, S. A cross-layer, mobility, and congestion-aware routing protocol for UAV networks. *IEEE Trans. Aerosp. Electron. Syst.* **2022**, *59*, 3778–3796. [CrossRef]
29. Feng, Y.; Wang, Y.; Zhang, B.; Zhou, L. Deep Reinforcement Learning Based TCP Congestion Control in UAV Assisted Wireless Networks. In Proceedings of the 2023 International Conference on Wireless Communications and Signal Processing (WCSP), Hangzhou, China, 2–4 November 2023; pp. 862–867.
30. Kang, M.; Jeon, S.W. Energy-efficient data aggregation and collection for multi-UAV-enabled IoT networks. *IEEE Wirel. Commun. Lett.* **2024**, *13*, 1004–1008. [CrossRef]
31. Hamissi, A.; Dhraief, A. A survey on the unmanned aircraft system traffic management. *ACM Comput. Surv.* **2023**, *56*, 1–37. [CrossRef]

32. Naqvi, H.A.; Hilman, M.H.; Anggorojati, B. Implementability improvement of deep reinforcement learning based congestion control in cellular network. *Comput. Netw.* **2023**, *233*, 109874. [CrossRef]
33. Li, Y.; Zhang, Z.; Sun, Q.; Huang, Y. A distributed framework for multiple UAV cooperative target search under dynamic environment. *J. Frankl. Inst.* **2024**, *361*, 106810. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.