Predictive Importance Sampling Based Coverage Verification for Multi-UAV Trajectory Planning

A dissertation submitted in
partial fulfilment for the degree of

Master of Technology

in

Computer Science

by

Snehashish Ghosh
Roll no. – **CS2431**

under the supervision of

Dr. Sasthi C. Ghosh
Advanced Computing and Microelectronics Unit (ACMU)

INDIAN STATISTICAL INSTITUTE, KOLKATA

June, 2026

This work has been accepted for presentation at the IEEE International Black Sea Conference on Communications and Networking (IEEE BlackSeaCom 2026), Bucharest, Romania.

* Preprint available at `https://arxiv.org/abs/2603.01687`

* Awarded student travel grant by Indian Statistical Institute for conference participation.

Certificate

This is to certify that the dissertation titled *"Predictive Importance Sampling Based Coverage Verification for Multi-UAV Trajectory Planning"* submitted by Snehashish Ghosh to the Indian Statistical Institute, Kolkata, fulfills the requirements for the degree of Master of Technology in Computer Science. The work presented in this dissertation is an authentic and original contribution conducted under my supervision and guidance. I confirm that this dissertation adheres to all the necessary academic standards and regulations of the institute.

Sasthi Charan Ghosh

Dr. Sasthi C. Ghosh
Advanced Computing and Microelectronics Unit (ACMU)
Indian Statistical Institute
Kolkata – 700108
India

Acknowledgement

I would like to express my sincere gratitude to Dr. Sasthi C. Ghosh, my advisor at the Advanced Computing and Microelectronics Unit of the Indian Statistical Institute, Kolkata. His expert guidance, constant encouragement, and insightful suggestions throughout this research have been invaluable. His deep expertise in wireless networks and stochastic optimization has been a great source of inspiration and direction.

I am grateful to all the faculty members and research scholars at the Indian Statistical Institute for their helpful discussions and academic support. Their feedback and perspectives have enriched this research considerably.

I also thank the administration and staff of the Indian Statistical Institute, Kolkata, for providing the necessary computational resources and infrastructure that made this work possible.

Finally, I wish to thank my family and friends for their unwavering support and motivation throughout my academic journey.

Declaration

I, Snehashish Ghosh, with Roll No. CS2431, hereby declare that the material presented in the dissertation titled *Predictive Importance Sampling Based Coverage Verification for Multi-UAV Trajectory Planning* represents original work carried out by me for the degree of Master of Technology in Computer Science at the Indian Statistical Institute, Kolkata.

Furthermore, I affirm that no sections of this report have been sourced or copied from external references without proper attribution. I am aware that any instances of plagiarism or the use of unacknowledged materials from third parties will be treated with the utmost seriousness.

Snehashish Ghosh.

Snehashish Ghosh
M.Tech (CS), CS2431
Indian Statistical Institute

Abstract

In next-generation wireless networks, unmanned aerial vehicle (UAV) networks are emerging as a promising solution for ultra-reliable low-latency communication (URLLC). A key challenge in millimeter-wave UAV networks is ensuring that mobile users are always in line-of-sight (LoS) coverage, since the current snapshot-based trajectory planning approach does not consider the mobility of the users during the decision interval, resulting in disastrous LoS gaps. For continuous coverage verification, standard uniform sampling is too computationally expensive, as it would need a large number of samples to estimate rare failure events that have latencies that are not suitable for real-time requirements.

In this work, we introduce a *Predictive Importance Sampling* (PIS) framework that significantly decreases sample complexity by focusing verification efforts on regions where failure is predicted. Specifically, we propose a Long Short-Term Memory Mixture Density Network (LSTM-MDN) architecture to learn multimodal user trajectory distributions and introduce a defense approach based on mixture sampling to handle the robustness against the prediction error. We show that PIS yields unbiased failure probability estimates that have lower variance than uniform sampling. We then combine PIS with Multi-Agent Deep Deterministic Policy Gradient (MADDPG) to perform coordinated multi-UAV trajectory planning based on an energy-aware multi-objective reward function that balances throughput, coverage, fairness and energy consumption. Based on the simulation results, our proposed method improves the coverage rate, throughput and verification latency in comparison with three state-of-the-art methods, thus enabling proactive coverage management for URLLC-aware UAV networks.

Keywords: Importance Sampling, Coverage Verification, URLLC, LSTM-MDN, Coordinated Trajectory Planning, UAV Networks, MADDPG.

Contents

Certificate i

Acknowledgement ii

Declaration iii

Abstract iv

1 Introduction 1
- 1.1 Background and Motivation 1
- 1.2 Problem Statement 1
- 1.3 Literature Survey 2
- 1.4 Proposed Approach and Contributions 3
- 1.5 Organization of the Report 4

2 Background and Related Theory 5
- 2.1 Ultra-Reliable Low-Latency Communication (URLLC) 5
- 2.2 Millimeter Wave (mmWave) Propagation 5
- 2.3 Monte Carlo Methods and Importance Sampling 5
  - 2.3.1 Standard Monte Carlo Estimation 5
  - 2.3.2 Importance Sampling 6
- 2.4 Long Short-Term Memory Networks 6
- 2.5 Mixture Density Networks 6
- 2.6 Multi-Agent Deep Deterministic Policy Gradient 7

3 System Model 8
- 3.1 Network Topology 8
- 3.2 Channel Model 8
- 3.3 User Traffic Classification 9
- 3.4 Mobility Model 9

4 Proposed Methodology 10
- 4.1 Rare Event Formulation 10
- 4.2 Proposed PIS Model 10
  - 4.2.1 Optimal Proposal Distribution 10
  - 4.2.2 Defensive Mixture Strategy 11
- 4.3 Theoretical Guarantees 11

4.4 LSTM-MDN Architecture for Trajectory Prediction 12
4.5 PIS Failure Probability Estimation 13
4.6 Coordinated Multi-UAV Trajectory Planning 13
4.6.1 Multi-Objective Reward Function 13
4.6.2 Algorithm 14
4.6.3 Complexity Analysis 15

5 Simulation Results **16**
5.1 Simulation Setup 16
5.2 Convergence and Throughput Analysis 17
5.3 Multi-Objective Normalized Performance 18
5.4 Quantitative Performance Comparison 18
5.5 Ablation Study: Effect of PIS Parameters 19

6 Conclusion and Future Work **20**
6.1 Conclusion 20
6.2 Future Work 20

Bibliography **22**

List of Figures

1.1 Analysis of coverage gaps during user movement. A user may have LoS connectivity at both the start and end of a decision interval Δt, but experience NLoS at intermediate positions due to obstacles, causing dropped connections. 2

5.1 Throughput analysis: convergence behavior and steady-state distribution. 17

5.2 Normalized performance comparison across four metrics: throughput, URLLC coverage ratio (CR), eMBB CR, and Jain's fairness index [29]. 18

List of Tables

5.1 Simulation Parameters 16
5.2 Performance Comparison 19
5.3 Effect of PIS Parameters 19

Chapter 1

Introduction

1.1 Background and Motivation

Unmanned aerial vehicle (UAV) networks have emerged as a promising platform for supporting ultra-reliable low-latency communication (URLLC) applications, including emergency services, industrial IoT and autonomous vehicles, as well as advanced mobile broadband (eMBB) applications, such as high throughput data-intensive applications [1,2]. eMBB users can accept some latency, but URLLC users demand sub-millisecond latency and very high reliability, which necessitates a pro-active rather than reactive management of the network. Over short distances, millimeter-wave (mmWave) can provide high data speeds and dependability. However, due to substantial propagation and penetration losses of mmWave, line-of-sight (LoS) conditions are the dominant factor in link quality, so maintaining a steady LoS connection is vital for users on the move [3].

There is a large body of work on optimal UAV placement [5] and trajectory planning [4] to achieve maximum coverage. In recent years, deep reinforcement learning, such as Multi-Agent Deep Deterministic Policy Gradient (MADDPG) [7,14] have been employed to coordinate multiple UAV trajectories. These approaches provide effective trajectory planning, but their coverage verification approaches have significant limitations that drive this work.

1.2 Problem Statement

Current coverage verification methods have a common problem: they can only verify coverage at the time points, and do not consider what is happening when the user is moving between decision intervals Δt. This leaves a critical gap that a user with LoS at time t and $t + \Delta t$ can have NLoS at some time in between if there are obstacles in the way when the user moves between them, as shown in Figure 1.1. These techniques also do not have predictive capabilities to predict the failure sites.

In other words, if a user has LoS connectivity with the UAV at the beginning and end of a time interval, it does not necessarily imply that the user will have LoS connectivity during the whole time interval. This results in the loss of connections, which is detrimental for delay-sensitive URLLC mission-critical applications and is

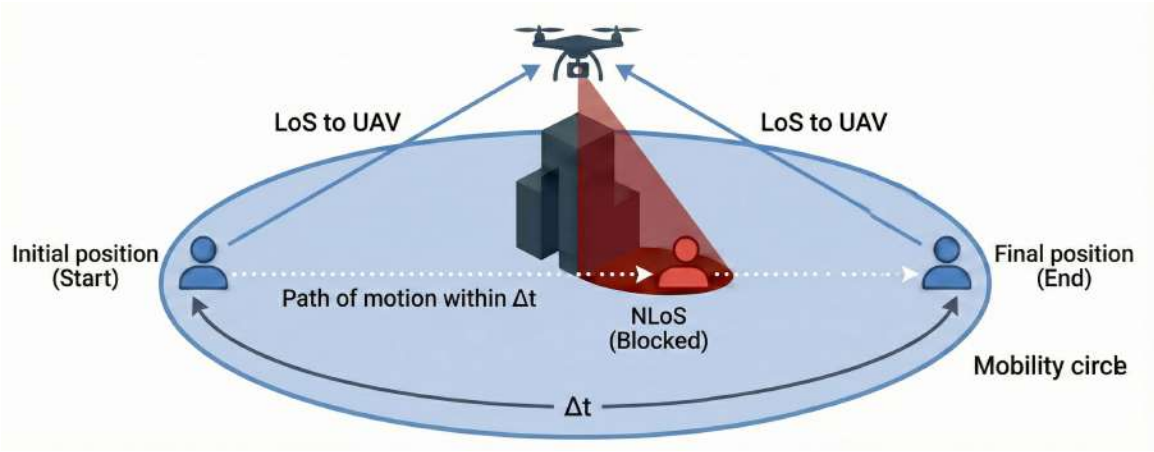


Figure 1.1: Analysis of coverage gaps during user movement. A user may have LoS connectivity at both the start and end of a decision interval Δt, but experience NLoS at intermediate positions due to obstacles, causing dropped connections.

also making the system unreliable for eMBB users.

A simple solution is to monitor the position of the users at very short time intervals, that is, make Δt very small. But if location updates were to be sent, they would occupy a considerable amount of data communication time, and significantly reduce user throughput.

In a statistical sense, the number of samples needed for a standard uniform Monte Carlo estimation is $N = \mathcal{O}(1/\sqrt{P_f})$ samples [8] which for $P_f = 10^{-4}$ requires about 10^6 samples. If each sample requires approximately $1\,\mu$s of ray-casting, then the latencies range from 1–100 ms per user which is far too high for URLLC requirements.

1.3 Literature Survey

The literature covers the aspects of UAV deployment, coverage verification and trajectory prediction:

- **Snapshot-based coverage verification.** The SINR thresholds and queue-based URLLC constraints are used for snapshot-based coverage verification by Qin *et al.* [11] only at discrete time slots under quasi-static assumptions. Dhinesh Kumar *et al.* [13] present a binary coverage indicator which verifies the data rates at certain checkpoints, and obtain good snapshot-based verification. In both cases, however, user movement in decision intervals is disregarded.

- **Deep reinforcement learning for UAV coordination.** A popular multi-agent actor-critic approach is MADDPG [14], which solves the problem of non-stationarity in the environment by adding policies of other agents to the centralized critic. In SAC-based approaches [11, 28] both the UAV trajectory and beamforming are optimized. These methods do not have effective continuous coverage verification.

- **Trajectory prediction.** The sequential past mobility data can be effectively modeled by the Long Short-Term Memory (LSTM) networks [16]. Mixture Density Networks (MDNs) [17] are networks that predict multiple trajectories for user movement along with their corresponding probabilities, capturing the inherent multimodality of user movement. The social context is used for crowd trajectory prediction in social LSTM [18] and DESIRE [19].

- **Importance Sampling.** Importance Sampling (IS) [9, 15] is a classical Variance Reduction (VR) technique which focuses the samples in the high-probability failure regions without sacrificing the unbiasedness of the estimator, suitable for rare-event estimation.

To our best knowledge, no previous work has tackled all three of the following problems: (1) How to efficiently check coverage for continuous movement between timesteps, (2) How to integrate learned path prediction with formal IS theory, and (3) How to coordinate multiple UAVs with reduced variance guarantees in real-time.

1.4 Proposed Approach and Contributions

To build on the gap above, we propose a framework called Predictive Importance Sampling (PIS) that provably reduces Monte Carlo variance and verification latency by directing computational efforts towards the most important regions. Specifically, our contributions are:

1. We develop a theoretically-grounded PIS framework for efficient coverage verification, and prove that it provides unbiased failure probability estimates with reduced variance versus uniform sampling under specified conditions. Unbiasedness ensures accurate URLLC reliability guarantees, while variance reduction enables real-time verification with significantly fewer samples.

2. We create an LSTM-MDN architecture as the predictive foundation that generates multi-modal trajectory distributions focusing sampling on anticipated failure locations, employing a defensive mixture approach combining prediction with uniform sampling backup for robustness.

3. We develop an adaptive multi-objective reward function with learnable weights that dynamically balances throughput, coverage, fairness, and energy efficiency based on network state.

4. We integrate PIS with MADDPG to design coordinated multi-UAV trajectory planning enabling real-time proactive coverage management, and establish our strategy's superiority over state-of-the-art approaches through comprehensive experiments.

1.5 Organization of the Report

Chapter 2 provides background on URLLC, mmWave channels, Monte Carlo methods, LSTM-MDN, and MADDPG. Chapter 3 describes the system model. Chapter 4 presents the PIS framework with theoretical proofs, the LSTM-MDN architecture, and the multi-UAV planning algorithm. Chapter 5 provides simulation results. Chapter 6 concludes with future directions.

Chapter 2

Background and Related Theory

2.1 Ultra-Reliable Low-Latency Communication (URLLC)

URLLC is a key service category in 5G and beyond wireless systems, targeting mission-critical applications such as industrial automation, autonomous vehicles, remote surgery, and emergency services. URLLC mandates end-to-end latency below 1 ms and packet success rate of 99.999% or higher [22]. These stringent requirements fundamentally differ from conventional broadband services and demand proactive, rather than reactive, network management.

2.2 Millimeter Wave (mmWave) Propagation

Millimeter-wave communications, operating at 30–300 GHz, offer enormous bandwidth but suffer from severe propagation characteristics including high path loss, atmospheric absorption, and susceptibility to blockage [23]. The 73 GHz band used in this work provides high capacity over short ranges but relies critically on LoS conditions. The path loss (in dB) for an air-to-ground link is [10]:

$$PL_{i,j}(d_{i,j}) = \alpha_{pl} + 10\beta_{pl} \log_{10}(d_{i,j}), \tag{2.1}$$

where α_{pl} and β_{pl} are environment-specific constants and $d_{i,j}$ is the Euclidean distance. Small-scale fading follows Rician (LoS) or Rayleigh (NLoS) distributions [24].

2.3 Monte Carlo Methods and Importance Sampling

2.3.1 Standard Monte Carlo Estimation

Monte Carlo (MC) methods estimate $\mu = \mathbb{E}_p[f(X)]$ by drawing N i.i.d. samples $X_i \sim p(x)$:

$$\hat{\mu}_{\mathrm{MC}} = \frac{1}{N} \sum_{i=1}^{N} f(X_i). \tag{2.2}$$

The estimator is unbiased with variance $\mathrm{Var}_p(f(X))/N$. For rare-event estimation where $\mu \ll 1$ (e.g., NLoS failure probability), the variance is large and N must be enormous for accuracy.

2.3.2 Importance Sampling

IS [9, 15] changes the sampling distribution while restoring the original expectation through importance weights. For any proposal $q(x)$ whose support contains that of $p(x)$:

$$\mu = \mathbb{E}_p[f(X)] = \mathbb{E}_q\left[f(X)\frac{p(X)}{q(X)}\right]. \tag{2.3}$$

The IS estimator $\hat{\mu}_q = \frac{1}{N}\sum_{i=1}^{N} f(X_i)w(X_i)$, with $X_i \sim q$ and $w(x) = p(x)/q(x)$, is unbiased. The optimal proposal minimizing variance is $q^*(x) \propto f(x)p(x)$, which concentrates samples where f is large.

2.4 Long Short-Term Memory Networks

LSTM networks [16] are a class of recurrent neural networks designed to model long-range dependencies in sequential data. Gating mechanisms — input gate i_t, forget gate f_t, and output gate o_t — selectively retain or discard information over time:

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f), \qquad i_t = \sigma(W_i[h_{t-1}, x_t] + b_i), \quad [16] \tag{2.4}$$

$$\tilde{c}_t = \tanh(W_c[h_{t-1}, x_t] + b_c), \qquad c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t, \quad [16] \tag{2.5}$$

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o), \qquad h_t = o_t \odot \tanh(c_t) \quad [16]. \tag{2.6}$$

where $\sigma(\cdot)$ is the sigmoid function and $\odot$ denotes elementwise multiplication. LSTMs are widely used for mobility modeling and trajectory prediction [18].

2.5 Mixture Density Networks

MDNs [17] combine neural networks with Gaussian Mixture Models (GMMs) to produce flexible, multimodal output distributions. Rather than predicting a single point, an MDN outputs GMM parameters:

$$p(y|x) = \sum_{k=1}^{K} \pi_k(x)\mathcal{N}(y \mid \mu_k(x), \Sigma_k(x)), \quad [17] \tag{2.7}$$

where $\pi_k(x)$ are mixture weights (summing to 1). MDNs capture multimodal behavior such as a user at an intersection who can turn left, right, or go straight — averaging these paths yields physically unrealistic predictions, whereas an MDN models each mode explicitly.

2.6 Multi-Agent Deep Deterministic Policy Gradient

MADDPG [14] is a multi-agent RL algorithm based on Centralized Training with Decentralized Execution (CTDE). Each agent i has an actor $\mu_i(s_i; \theta_i^\mu)$ mapping local observations to actions, and a centralized critic $Q_i(s_{1:N}, a_{1:N}; \theta_i^Q)$ using global state and all agents' actions for training. The actor gradient is:

$$\nabla_{\theta_i^\mu} J \approx \mathbb{E}\Big[\nabla_{\theta_i^\mu} \mu_i(s_i)\, \nabla_{a_i} Q_i(s_{1:N}, a_{1:N})\big|_{a_i=\mu_i(s_i)}\Big]\,. \tag{2.8}$$

Centralized training enables cooperative behavior, while decentralized execution requires no inter-agent communication at deployment time.

Chapter 3

System Model

3.1 Network Topology

We consider a three-dimensional air-to-ground cellular wireless communication system with N_{UAV} UAVs acting as aerial base stations and $N_{\textbf{users}}$ ground users over a geographical area of $a \times a\,\text{m}^2$. Both UAVs and users are mobile. The environment contains M static obstacles (buildings) that can block LoS paths between UAVs and ground users.

Each UAV operates within an altitude range of h_{min} to h_{max} meters with maximum speed $v_{\text{UAV}}\,\text{m/s}$, ensuring safe flight and collision avoidance within the designated green zone. Ground users are assumed to be at zero altitude. Time is discretized into decision intervals of duration Δt seconds.

3.2 Channel Model

The path loss (in dB) for the link between UAV i and user j follows (2.1), recalled here for convenience:

$$PL_{i,j}(d_{i,j}) = \alpha_{pl} + 10\beta_{pl}\log_{10}(d_{i,j}). \tag{3.1}$$

The received power (in dBm) at user j from UAV i is:

$$P_{i,j} = P_i + G_i + G_j - PL_{i,j}(d_{i,j}), \tag{3.2}$$

where P_i is the transmit power and G_i, G_j are antenna gains. The achievable throughput from Shannon's capacity formula [25] is:

$$T_{i,j} = B_w \log_2\left(1 + \frac{P_{i,j} \cdot |g_{i,j}|^2}{\sigma_0^2}\right), \tag{3.3}$$

where B_w is the channel bandwidth, σ_0^2 is the AWGN power, and $|g_{i,j}|$ is the small-scale fading coefficient following Rician (LoS) or Rayleigh (NLoS) distributions [24].

3.3 User Traffic Classification

Users are classified into two service classes based on delay tolerance:

- **URLLC users** require sub-millisecond latency and very high reliability. Even minor coverage interruptions can cause catastrophic failures. URLLC users require continuous LoS coverage verification throughout their mobility region during decision interval Δt. A user is considered covered if the estimated failure probability $\hat{P}_f$ is below a pre-defined threshold ϵ.
- **eMBB users** are designed for high-throughput, data-intensive applications that can tolerate minor delays. They do not require extensive LoS coverage verification; coverage is assessed by whether data rate exceeds a threshold at the end of each interval.

3.4 Mobility Model

User velocities are bounded by $v_u \in [0, v_{\max}]$ m/s, representing pedestrian and vehicular mobility. Users remain within the simulation area through boundary reflection.

At a particular time $t \in [0, \Delta t]$, a user at position u_t with velocity v_u can occupy any position within the *mobility circle*:

$$\mathcal{C} = \{z \in \mathbb{R}^2 : \|z - u_t\| \leq r\}, \quad r = v_u \cdot \Delta t. \tag{3.4}$$

UAVs execute continuous control actions $\Delta \mathbf{p} = (\Delta x, \Delta y, \Delta z)$ generated by actor networks, with position updates:

$$p_{t+1} = p_t + v_{\text{UAV}} \cdot \Delta t, \tag{3.5}$$

subject to hard boundary constraints ensuring UAVs remain within the safe flight zone.

Chapter 4

Proposed Methodology

This chapter presents our Predictive Importance Sampling (PIS) framework. We first formulate the rare-event challenge, derive the PIS model, prove its theoretical guarantees, describe the LSTM-MDN architecture, and present the MADDPG-based multi-UAV planning algorithm.

4.1 Rare Event Formulation

Unlike eMBB, URLLC mandates continuous coverage throughout $\mathcal{C}$. Since verifying uncountably infinite points is intractable, we estimate the failure probability. Assuming worst-case uniform uncertainty over $\mathcal{C}$, the failure probability is:

$$P_f = \frac{\int_{\mathcal{C}} \mathbb{I}_F(z)\, dz}{\int_{\mathcal{C}} dz} = \mathbb{E}_p[\mathbb{I}_F(Z)], \tag{4.1}$$

where $Z \sim \text{Uniform}(\mathcal{C})$, $p(z) = 1/A$ with A being the area of $\mathcal{C}$, and $\mathbb{I}_F(z) = 1$ if position z is in NLoS (failure), 0 otherwise. The standard Monte Carlo estimator is:

$$\hat{P}_f^{\textbf{uniform}} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}_F(Z_i), \quad Z_i \sim \text{Uniform}(\mathcal{C}). \tag{4.2}$$

This suffers from the *rare event challenge*: NLoS events are rare in well-planned deployments, making uniform sampling grossly inefficient.

4.2 Proposed PIS Model

4.2.1 Optimal Proposal Distribution

From IS theory (Section 2.3), the optimal proposal minimizing variance is $q^*(z) \propto \mathbb{I}_F(z)p(z) = \mathbb{I}_F(z)/A$, i.e., it concentrates samples on the NLoS failure set $\mathcal{F} = \{z \in \mathcal{C} : \mathbb{I}_F(z) = 1\}$. Our goal is to construct a practical approximation $q(z) \approx q^*(z)$.

4.2.2 Defensive Mixture Strategy

A naive learned proposal $q_{\textbf{pred}}(z)$ can assign near-zero probability to some failure regions (when the predictor errs), causing importance weights $w(z) = p(z)/q(z) \to \infty$ and catastrophic variance inflation. To prevent this, we employ a *defensive mixture* [26, 27]:

$$q_{\text{PIS}}(z) = \alpha \cdot q_{\textbf{pred}}(z) + (1-\alpha) \cdot p(z), \quad \alpha \in [0, 1), \tag{4.3}$$

where $q_{\textbf{pred}}(z)$ is the LSTM-MDN prediction (Section 4.4) concentrating on predicted failure regions, and $p(z) = 1/A$ is the uniform fallback. Since $p(z) > 0$ everywhere in $\mathcal{C}$:

$$q_{\text{PIS}}(z) \geq \frac{1-\alpha}{A} > 0 \quad \forall\, z \in \mathcal{C}, \tag{4.4}$$

guaranteeing bounded importance weights $w(z) \leq A/(1-\alpha)$.

The parameter α provides tunable control: higher α (close to 1) implies greater reliance on prediction; lower α (close to 0) prioritizes robust uniform coverage. The value of α is chosen empirically to balance both objectives.

4.3 Theoretical Guarantees

Theorem 4.1 (Unbiasedness of PIS Estimator). *The PIS estimator*

$$\hat{P}_f^{PIS} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}_F(Z_i)\frac{p(Z_i)}{q_{PIS}(Z_i)}, \quad Z_i \overset{\text{i.i.d.}}{\sim} q_{PIS}, \tag{4.5}$$

is unbiased: $\mathbb{E}_{q_{PIS}}[\hat{P}_f^{PIS}] = P_f$.

Proof. By linearity of expectation and the i.i.d. property of samples:

$$\begin{aligned} \mathbb{E}_{q_{\text{PIS}}}[\hat{P}_f^{\text{PIS}}] &= \mathbb{E}_{q_{\text{PIS}}}\left[\mathbb{I}_F(Z)\frac{p(Z)}{q_{\text{PIS}}(Z)}\right] = \int_{\mathcal{C}} \mathbb{I}_F(z)\frac{p(z)}{q_{\text{PIS}}(z)} \cdot q_{\text{PIS}}(z)\, dz \\ &= \int_{\mathcal{C}} \mathbb{I}_F(z)p(z)\, dz = \mathbb{E}_p[\mathbb{I}_F(Z)] = P_f, \end{aligned} \tag{4.6}$$

where $q_{\text{PIS}}(z)$ terms cancel because $q_{\text{PIS}}(z) > 0$ for all $z \in \mathcal{C}$. □

Theorem 4.2 (Variance Reduction). *Let* $\mathcal{F} = \{z \in \mathcal{C} : \mathbb{I}_F(z) = 1\}$ *be the failure region. If* $q_{PIS}(z) \geq p(z)$ *for all* $z \in \mathcal{F}$, *then:*

$$\text{Var}_{q_{PIS}}(\hat{P}_f^{PIS}) \leq \text{Var}_p(\hat{P}_f^{uniform}). \tag{4.7}$$

Proof. The variance of the uniform estimator, using $\mathbb{I}_F(Z)^2 = \mathbb{I}_F(Z)$ for binary indicators, is:

$$\text{Var}_p(\hat{P}_f^{\textbf{uniform}}) = \frac{1}{N}\left[\mathbb{E}_p[\mathbb{I}_F(Z)^2] - P_f^2\right] = \frac{P_f(1-P_f)}{N}. \tag{4.8}$$

The variance of the PIS estimator is:

$$\mathrm{Var}_{q_{\mathrm{PIS}}}(\hat{P}_f^{\mathrm{PIS}}) = \frac{1}{N}\left[\mathbb{E}_{q_{\mathrm{PIS}}}\left[(\mathbb{I}_F(Z)w(Z))^2\right] - P_f^2\right]. \tag{4.9}$$

For variance reduction it suffices to show $\mathbb{E}_{q_{\mathrm{PIS}}}[(\mathbb{I}_F w)^2] \le P_f$. Converting:

$$\mathbb{E}_{q_{\mathrm{PIS}}}[(\mathbb{I}_F w)^2] = \int_{\mathcal{C}} \mathbb{I}_F(z)\frac{p(z)^2}{q_{\mathrm{PIS}}(z)}\,dz = \mathbb{E}_p\left[\mathbb{I}_F(Z)\frac{p(Z)}{q_{\mathrm{PIS}}(Z)}\right]. \tag{4.10}$$

Under the assumption $q_{\mathrm{PIS}}(z) \ge p(z)$ for all $z \in \mathcal{F}$, we have $p(z)/q_{\mathrm{PIS}}(z) \le 1$ on $\mathcal{F}$, and $\mathbb{I}_F(z) = 0$ off $\mathcal{F}$. Hence pointwise:

$$\mathbb{I}_F(z)\frac{p(z)}{q_{\mathrm{PIS}}(z)} \le \mathbb{I}_F(z) \quad \forall\, z \in \mathcal{C}. \tag{4.11}$$

Taking the expectation under p gives $\mathbb{E}_p[\mathbb{I}_F(Z)p(Z)/q_{\mathrm{PIS}}(Z)] \le P_f$, completing the proof. □

Design Principle. Theorems 4.1–4.2 jointly state that the predictive model $q_{\mathbf{pred}}$ must push sufficient probability mass onto failure regions so that $q_{\mathrm{PIS}}(z) \ge p(z)$ on $\mathcal{F}$. If the predictor is accurate, variance reduction is guaranteed. If it errs, variance may exceed uniform sampling but remains bounded and controllable via α.

4.4 LSTM-MDN Architecture for Trajectory Prediction

To determine the optimal proposal distribution $q^* \propto I_F(z)$, we must accurately predict user locations and NLoS regions by modeling their historical movement. User movement exhibits strong trends. At time t, we observe H sequences of historical velocity data $\{(\mathbf{v}_{x,t-H}, \mathbf{v}_{y,t-H}), \ldots, (\mathbf{v}_{x,t-1}, \mathbf{v}_{y,t-1})\}$ and feed them into a 2-layer LSTM with hidden dimension 128, capturing movement patterns in the final hidden state h_t. Standard networks predict a single trajectory, but real-world user movement is multimodal with multiple possible directions. For example, a user at a street corner can turn left, right, or continue straight, averaging these yields physically unrealistic predictions (e.g., walking into a building).

MDNs [17] solve this by outputting Gaussian mixture model (GMM) parameters instead of a single prediction. The MDN computes K possible paths using LSTM state h_t: mixture weights $\pi_k(h_t)$ (summing to 1), mean positions $\boldsymbol{\mu}_k(h_t) \in \mathbb{R}^2$, and covariances $\boldsymbol{\Sigma}_k(h_t) \in \mathbb{R}^{2\times 2}$. The predictive distribution is:

$$q_{\mathbf{pred}}(z|h_t) = \sum_{k=1}^{K} \pi_k(h_t)\mathcal{N}(z|\boldsymbol{\mu}_k(h_t), \boldsymbol{\Sigma}_k(h_t)). \tag{4.12}$$

The model is trained using negative log-likelihood (NLL) loss. The architecture comprises an input layer (velocity sequences of size $H \times 2$), a 2-layer LSTM, and an MDN output head with softmax functions on weights π_k, linear functions on means $\boldsymbol{\mu}_k$, and exponential functions ensuring positive diagonal covariance.

4.5 PIS Failure Probability Estimation

We now present **Algorithm 1** which computes PIS failure probability. It begins by processing the user's velocity history $\mathcal{H}_v$ via the LSTM-MDN to generate trajectory predictions. A defensive mixture proposal $q_{pis}(z)$ is constructed to ensure sampling robustness. The core loop executes IS: it draws samples from q_{pis}, verifies LoS, and computes importance weights. The aggregated failure probability $\hat{P}_f$ is then thresholded against ϵ to output a binary coverage quality metric Q_u. This Q_u serves as the statistically verified ground truth for the reward function in the next stage.

Algorithm 1 PIS Failure Probability Estimation

Input: User state u_t, velocity history H_v, sample count N, mixture weight α
Output: Binary coverage quality Q_u

```
Predict: {π_k, μ_k, Σ_k} ← LSTM-MDN(H_v)
Construct proposal: q_PIS(z) = α Σ_k π_k N(z|μ_k, Σ_k) + (1 − α) U(C)
W_fail ← 0
for i = 1 to N do
    Sample z_i ∼ q_PIS(z)
    Reject if ||z_i − u_t|| > r    (enforce z_i ∈ C)
    Check LoS: compute I_F(z_i)
    if I_F(z_i) = 1 then
        w_i ← (1/A) / q_PIS(z_i)
        W_fail ← W_fail + w_i
    end if
end for
P̂_f ← W_fail/N
Q_u ← 1[P̂_f < ε]
return Q_u
```

Complexity analysis: Complexity of Algorithm 1 is $O(N \cdot C_{los}) \approx O(N)$, where N is the sample count and C_{los} is the cost of LoS check (typically $O(1)$ with height maps).

4.6 Coordinated Multi-UAV Trajectory Planning

4.6.1 Multi-Objective Reward Function

We design an adaptive multi-objective reward for proactive coverage maintenance:

$$R_t = \sum_{j \in \mathcal{R}} \omega_j \, \Phi_j(R_j) + \sum_{k \in \mathcal{P}} \omega_k \, \Phi_k(P_k), \tag{4.13}$$

where $\mathcal{R} = \{\text{throughput, coverage, load balance}\}$, $\mathcal{P} = \{\text{energy, collision}\}$, and $\Phi(\cdot)$ normalizes each component into a dimensionless utility ratio in $[0, 1]$.

Normalized throughput reward:

$$\Phi_{\textbf{thr}} = \frac{\sum_{(i,u)} B_w \log_2(1 + \text{SNR}_{i,u})}{N_{\textbf{users}} \cdot B_w \cdot \log_2(1 + \text{SNR}_{\max})}, \tag{4.14}$$

where $\text{SNR}_{\max} = P_{tx} \cdot g(h_{\min})/N_0$.

Normalized coverage reward:

$$\Phi_{\textbf{cov}} = \frac{\sum_u w_u Q_u}{\sum_u w_u}, \tag{4.15}$$

where Q_u is the binary coverage quality from Algorithm 1.

Normalized load balance reward:

$$\Phi_{\textbf{bal}} = \frac{\bar{n} - \sigma_n}{\bar{n} + \epsilon_0}, \tag{4.16}$$

where $\bar{n}$ is the mean user count per UAV and σ_n is its standard deviation.

Normalized energy consumption penalty:

$$\Phi_{\textbf{energy}} = 1 - \frac{\sum_i \|\Delta p_i\|}{N_{\text{UAV}} \cdot v_{\max} \cdot \Delta t}. \tag{4.17}$$

Normalized collision penalty:

$$\Phi_{\textbf{coll}} = \frac{1}{N_{\textbf{pairs}}} \sum_{i<j} \exp\left(-\frac{|d_{ij} - d_{\textbf{shift}}|}{k_{\textbf{scale}}}\right), \tag{4.18}$$

where $N_{\textbf{pairs}}$ is the number of UAV pairs and d_{ij} is the inter-UAV distance.

Rather than hand-tuning the weights ω_i, we employ a learnable weight network $\pi_\omega : \mathcal{S} \to \Delta^5$ (5-dimensional probability simplex) that dynamically adjusts priorities based on global network state.

4.6.2 Algorithm

Algorithm 2 discusses the integration of above computed reward function and uses centralized training decentralized execution (CTDE) for coordinated trajectory planning. At the start of an episode, the weight network π_ω analyzes the global state to adaptively set objective priorities ω (Line 3). During execution, Line 8 calls Algorithm 1 to compute Q_u for every user. This value feeds directly into the normalized coverage reward Φ_{cov} (Line 11), which is aggregated with throughput, load balance, and safety penalties into the global reward R_t. Coordination is enforced during the training phase (Line 15), where the centralized critic Q_i updates based on the *joint* actions of all UAVs, effectively learning to penalize non-cooperative behaviors like collisions or redundant coverage.

Algorithm 2 PIS-Guided Multi-UAV Trajectory Planning

Input: Agents $\mathcal{I}$, Environment $\mathcal{E}$, Weight Network π_ω, Actor μ, Critic Q

```
for episode ep = 1 to M do
    Observe joint state s = {s_1, ..., s_N}
    Get adaptive weights: ω ← π_ω(s_global)
    for step t = 1 to T do
        Select actions a_t = {μ_i(s_i) + N_t}_{i∈I}
        Execute actions: s' ← Step(a_t)
        for each UAV i ∈ I do
            Call Algorithm 1 for assigned URLLC users ⇒ Q_u
        end for
        Compute normalized components via Eqs. (4.14)–(4.18)
        Compute global reward: R_t ← Σ_j ω_j Φ_j + Σ_k ω_k Φ_k
        Store ⟨s, a, R_t, s'⟩ in replay buffer D
        Sample batch B from D
        Coordination: Update critic Q_i using joint actions a_{1:M} and S_global
        Update actors μ_i via deterministic policy gradient
        Update weight network π_ω to maximize long-term utility
        s ← s'
    end for
end for
```

4.6.3 Complexity Analysis

The computational complexity of Algorithm 2 is analyzed below. Let $J = |\mathcal{I}|$ denote the number of UAV agents, K_{URLLC} denote the number of URLLC users requiring PIS verification, and N denote the number of importance samples per user in Algorithm 1. The actor μ and critic Q networks have L_μ and L_Q layers respectively, where the l-th layer of the actor network has U_l neurons and the m-th layer of the critic network has V_m neurons. The weight network π_ω has $|\pi_\omega|$ parameters.

Therefore, the **training complexity** is $\mathcal{O}(M \cdot T \cdot (J^2 |\mathcal{B}| \sum_{l=1}^{L_\mu} U_{l-1} U_l + J|\mathcal{B}| \sum_{m=1}^{L_Q} V_{m-1} V_m + J \cdot K_{\text{URLLC}} \cdot N))$, where $|\mathcal{B}|$ is the mini-batch size.

The **deployment complexity** is $\mathcal{O}(T \cdot (J \sum_{l=1}^{L_\mu} U_{l-1} U_l + J \cdot K_{\text{URLLC}} \cdot N)) = \mathcal{O}(T(J|s||a| + J \cdot K_{\text{URLLC}} \cdot N))$, where $|s|$ and $|a|$ are the dimensions of state and action, respectively.

Chapter 5

Simulation Results

5.1 Simulation Setup

We compare the proposed MADDPG-PIS against three baselines:

1. **MADDPG** [14] — standard multi-agent actor-critic, no coverage prediction.
2. **SAC** [11, 28] — off-policy soft actor-critic jointly optimizing trajectories and beamforming.
3. **MADDPG-BO** [13] — MADDPG augmented with Bayesian optimization for hyperparameter tuning.
4. **Uniform-1000** — uniform sampling with $N = 1000$, used as an accuracy reference.

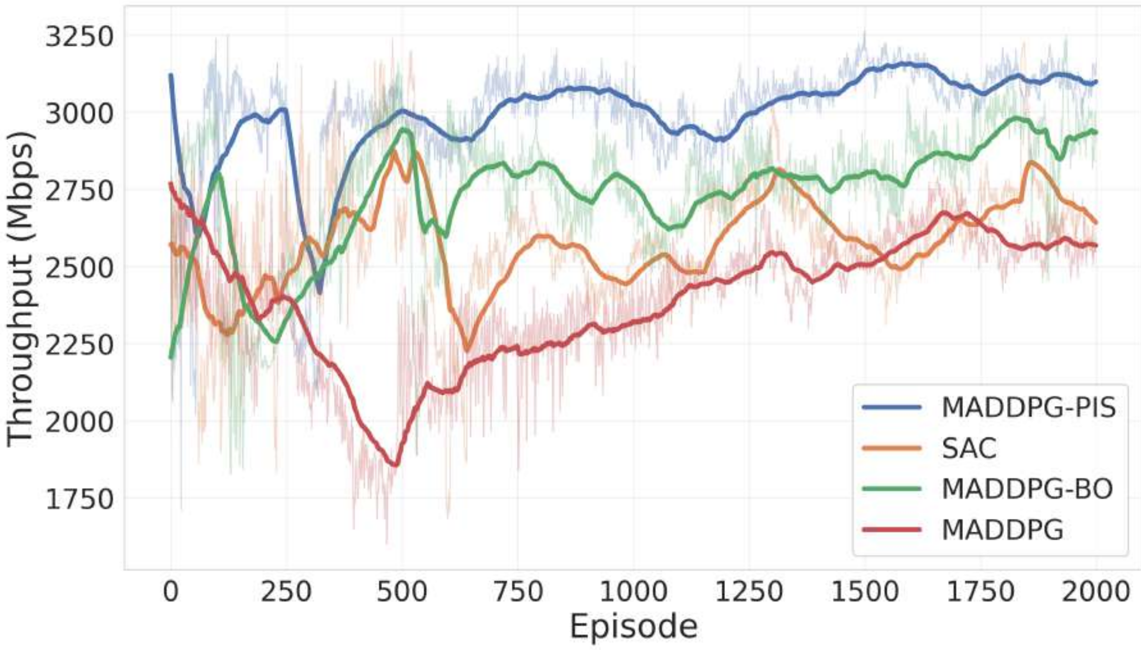
Simulation parameters are summarized in Table 5.1.

Table 5.1: Simulation Parameters

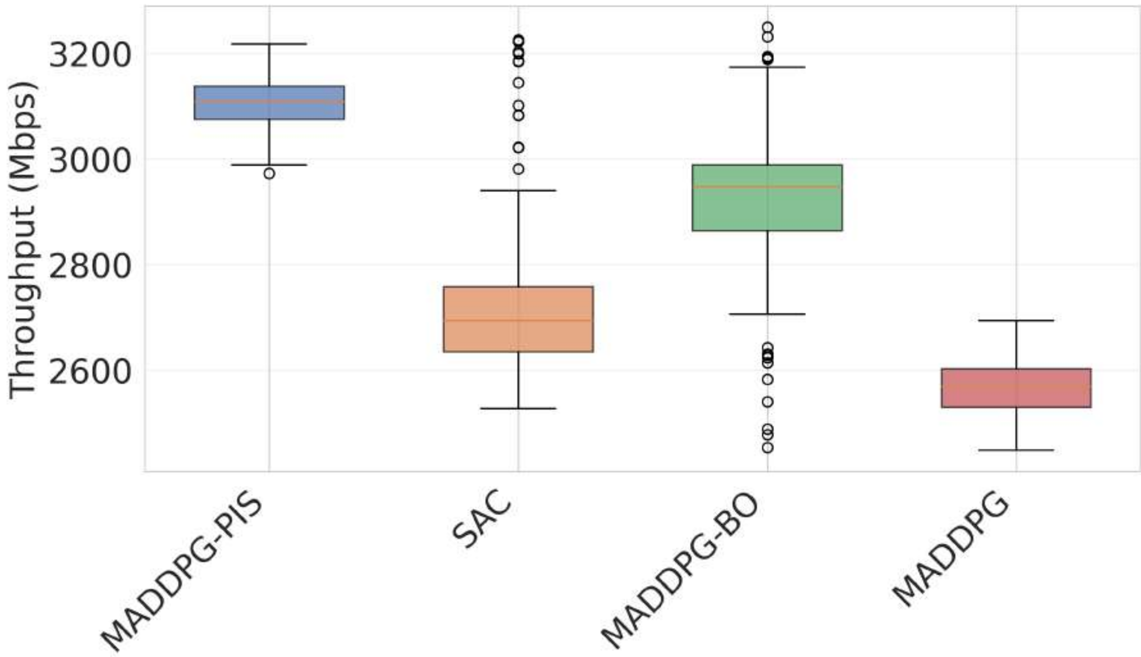
Parameter	**Value**
Simulation area ($a \times a\,\mathrm{m}^2$)	$1500 \times 1500\,\mathrm{m}$
Number of URLLC users	30
Number of eMBB users	200
Number of UAVs (N_{UAV})	5
UAV communication range	300 m
Decision interval (Δt)	10 s
Transmission power (P_i)	30 dBm [21]
UAV maximum speed ($v_{\max}$)	30 m/s [13]
Mixture weight (α)	0.6
LSTM hidden dimension	128
Replay buffer size	50 000
Rician fading factor	2 [11]
UAV altitude range	$[22, 150]$ m [20]
Carrier frequency	73 GHz [21]
LoS path-loss constants ($\alpha_{pl}, \beta_{pl}, \sigma_0$)	69.8, 2, 3.1 [21]

5.2 Convergence and Throughput Analysis

Figure 5.1 presents episode-wise throughput and steady-state distribution over the final 200 episodes.



(a) Episode-wise throughput (Mbps)



(b) Throughput distribution (final 200 episodes)

Figure 5.1: Throughput analysis: convergence behavior and steady-state distribution.

MADDPG-PIS converges to approximately 3100 Mbps by episode 1500, demonstrating superior asymptotic performance. MADDPG exhibits prolonged instability (episodes 0–500) before converging, whereas PIS-guided exploration accelerates early-stage learning. In the box-plot, MADDPG-PIS exhibits the tightest distribution (IQR = 62.89 Mbps), indicating consistent performance. In contrast, SAC, MADDPG-BO, and MADDPG have IQRs of 123.43, 124.57, and 73.76 Mbps respectively.

5.3 Multi-Objective Normalized Performance

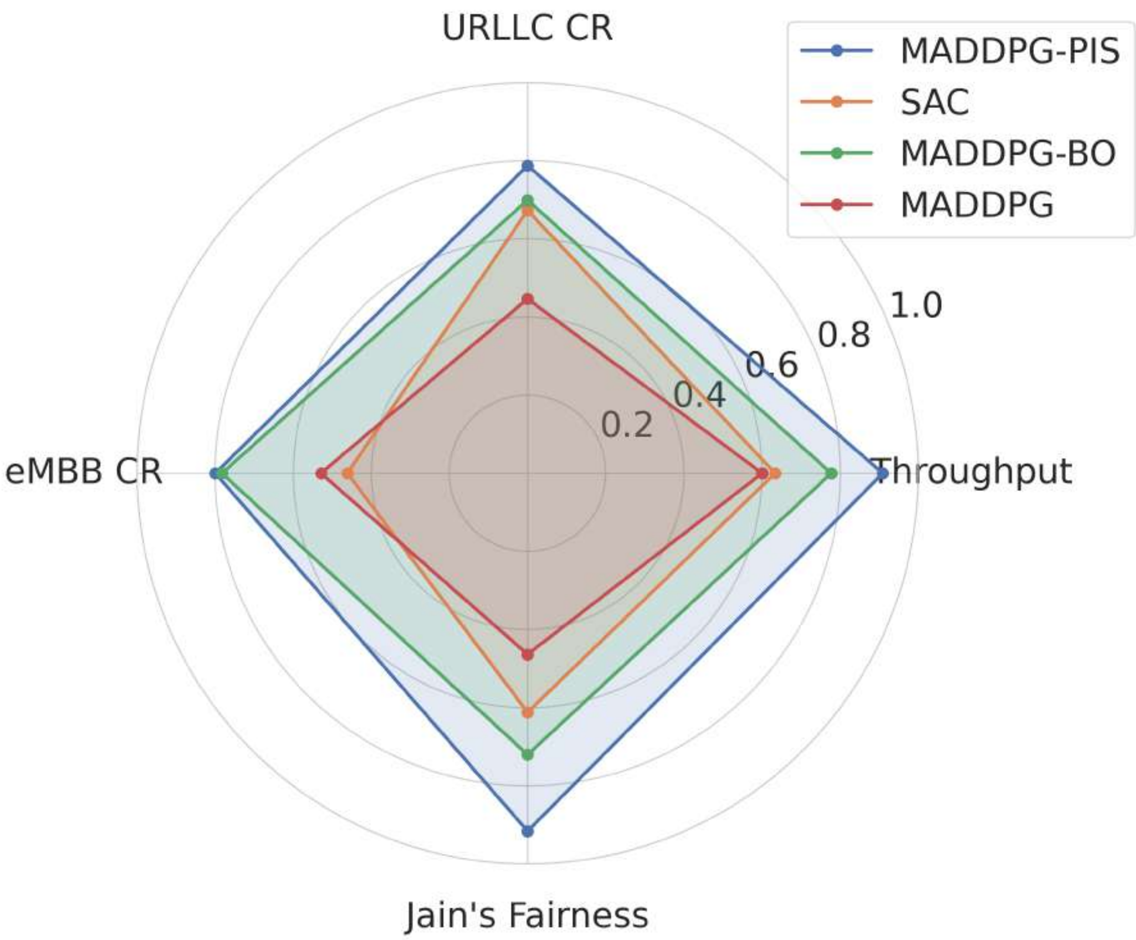


Figure 5.2: Normalized performance comparison across four metrics: throughput, URLLC coverage ratio (CR), eMBB CR, and Jain's fairness index [29].

The radar chart (Figure 5.2) normalizes four critical metrics to $[0, 1]$. MADDPG-PIS achieves the largest enclosed area, demonstrating balanced excellence across competing objectives. It dominates in throughput, URLLC coverage ratio, and Jain's fairness index. The rhomboid shape of vanilla MADDPG reveals its bias toward throughput at the expense of reliability.

5.4 Quantitative Performance Comparison

Table 5.2 reports absolute performance metrics over converged episodes.

MADDPG-PIS achieves 3127.85 Mbps throughput, corresponding to improvements of $\uparrow$22% over vanilla MADDPG, $\uparrow$14.8% over SAC, and $\uparrow$7.6% over MADDPG-BO. URLLC coverage rate of 74.1% represents a $\uparrow$10.3% gain over MADDPG-BO. The Jain's fairness index [29] of 0.71 is $\uparrow$16.4% higher than MADDPG-BO.

Table 5.2: Performance Comparison

Method	**Throughput (Mbps)**	**URLLC CR (%)**	**eMBB CR (%)**	**Energy (kJ)**	**Jain's Index** [29]
Proposed	**3127.85**	**74.1**	**61.7**	116.1	**0.71**
MADDPG-BO	2906.49	64.0	59.8	123.6	0.61
SAC	2724.62	63.8	51.5	130.3	0.59
MADDPG	2563.03	55.2	51.5	92.1	0.52

5.5 Ablation Study: Effect of PIS Parameters

Table 5.3 investigates the role of the mixture weight α and sample count N (By default $N = 100$).

Table 5.3: Effect of PIS Parameters

Configuration	Throughput (Mbps)	URLLC CR(%)	Var($\hat{P}_f$) ($\times 10^{-5}$)	Latency per user(ms)
Proposed ($\alpha = 0.6$)	**3097**	**72**	5.2	0.57
High α ($\alpha = 0.9$)	2219	50.6	29.8	0.42
Low α ($\alpha = 0.3$)	2376	52.8	19.1	0.64
Uniform ($N = 1000$)	2998	68.1	4.8	11

Key observations:

- $\alpha = 0.6, N = 100$ **(proposed):** Achieves the best balance of accuracy (5.2×10^{-5} variance) and latency (0.57 ms), confirming this as the sweet spot between prediction reliance and uniform fallback.
- $\alpha = 0.9, N = 100$ **(high):** Insufficient uniform fallback exposes prediction errors, inflating variance to 29.8×10^{-5} and degrading coverage to 50.6%.
- $\alpha = 0.3, N = 100$ **(low):** The uniform fallback dominates, under-utilizing predictions and yielding variance 19.1×10^{-5} with sub-optimal coverage.
- **Uniform** ($N = 1000$)**:** Achieves comparable accuracy (4.8×10^{-5}) but at 11 ms latency — 20× slower than PIS — rendering it incompatible with URLLC requirements.

The 10× sample reduction (100 vs. 1000) causes only a marginal variance increase, empirically demonstrating PIS's superior per-sample efficiency, consistent with Theorems 4.1–4.2.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

In this dissertation, a Predictive Importance Sampling (PIS) framework for sample-efficient continuous coverage verification in multi-UAV mmWave networks is presented. The main contributions and results are:

1. We found the key issue of snapshot based coverage verification: user movement inside decision intervals is not taken into consideration, which leads to NLoS gaps that are critical for URLLC applications but are not detected by the current approaches.

2. We introduced the PIS framework of the LSTM-MDN trajectory prediction and a defensive mixture IS strategy. In Theorems 4.1–4.2 we showed that PIS can give failure probability estimates that are unbiased, with provably lower variance, under a mild condition on the proposal distribution.

3. The framework can decrease the verification latency by around $20\times$ (from 11 ms to 0.57 ms per user) compared to uniform sampling with 1000 samples, where sub-millisecond latency is compatible with URLLC.

4. Compared with state-of-the-art baselines, integration with MADDPG by an adaptive multi-objective reward function leads to better performance in terms of throughput (| 22%), URLLC coverage rate (| 34.2%), and Jain's fairness index (| 36.5%) versus vanilla MADDPG.

6.2 Future Work

This work suggests a number of fruitful avenues to pursue:

- Online learning of the mixture weight α according to the prediction uncertainty of the mixture might further enhance efficiency and robustness.

- The current framework focuses on 2D ground-user mobility. It would be broadened if applied to 3D aerial user mobility (e.g. urban air mobility).

- The use of transformer-based models that explicitly model inter-user social interactions instead of the LSTM could boost the prediction accuracy of multimodal prediction.
- **Hardware validation:** deployment and validation of the framework on real UAV testbeds for practical performance under hardware constraints and real channel conditions.
- The PIS framework is not limited to UAV networks but can be used in any mobile wireless system that needs to verify spatial coverage at a low number of samples, including terrestrial heterogeneous networks or low-Earth orbit satellite systems.
- **Distributed implementation:** Distribute the PIS computation across UAVs to avoid centralized bottlenecks and scalability.

Bibliography

[1] M. Mozaffari, W. Saad, M. Bennis, Y.-H. Nam, and M. Debbah, "A tutorial on UAVs for wireless networks: Applications, challenges, and open problems," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2334–2360, 2019.

[2] W. Saad, M. Bennis, and M. Chen, "A vision of 6G wireless systems: Applications, trends, technologies, and open research problems," *IEEE Network*, vol. 34, no. 3, pp. 134–142, 2020.

[3] A. Al-Hourani, S. Kandeepan, and S. Lardner, "Optimal LAP altitude for maximum coverage," *IEEE Wireless Commun. Lett.*, vol. 3, no. 6, pp. 569–572, 2014.

[4] Y. Zeng and R. Zhang, "Energy-efficient UAV communication with trajectory optimization," *IEEE Trans. Wireless Commun.*, vol. 16, no. 6, pp. 3747–3760, 2017.

[5] M. Alzenad, A. El-Keyi, F. Lagum, and H. Yanikomeroglu, "3D placement of an unmanned aerial vehicle base station (UAV-BS) for energy-efficient maximal coverage," *IEEE Wireless Commun. Lett.*, vol. 6, no. 4, pp. 434–437, 2017.

[6] X. Yan, A. Sekercioglu, and S. Narayanan, "A survey of vertical handover decision algorithms in fourth generation heterogeneous wireless networks," *Comput. Networks*, vol. 54, pp. 1848–1863, 2010.

[7] D. Ravi and R. Arunachalam, "Optimizing UAV deployment for maximizing coverage and data rate efficiency using multi-agent deep deterministic policy gradient and Bayesian optimization," *Phys. Commun.*, vol. 69, p. 102621, 2025.

[8] R. Y. Rubinstein and D. P. Kroese, *Simulation and the Monte Carlo Method*, 3rd ed. Wiley, 2016.

[9] A. B. Owen, *Monte Carlo Theory, Methods and Examples*, 2013.

[10] M. R. Akdeniz, Y. Liu, M. K. Samimi, S. Sun, S. Rangan, T. S. Rappaport, and E. Erkip, "Millimeter wave channel modeling and cellular capacity evaluation," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1164–1179, 2014.

[11] P. Qin, Y. Fu, Z. Yu, J. Zhang, and X. Zhao, "URLLC-Aware trajectory plan and beamforming design for NOMA-aided UAV integrated sensing, communication, and computation networks," *IEEE Trans. Veh. Technol.*, vol. 74, pp. 1610–1625, 2025.

[12] T. Yang, C. H. Foh, F. Heliot, C. Y. Leow, and P. Chatzimisios, "Self-organization drone-based unmanned aerial vehicles (UAV) networks," in *Proc. ICC*, pp. 1–6, 2019.

[13] R. D. Kumar and A. Rammohan, "Optimizing UAV deployment for maximizing coverage and data rate efficiency using multi-agent deep deterministic policy gradient and Bayesian optimization," *Phys. Commun.*, vol. 69, p. 102621, 2025.

[14] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch, "Multi-agent actor-critic for mixed cooperative-competitive environments," in *Proc. NIPS*, pp. 6382–6393, 2017.

[15] R. Y. Rubinstein and D. P. Kroese, *Simulation and the Monte Carlo Method*, 3rd ed. Wiley, 2017.

[16] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[17] C. M. Bishop, "Mixture density networks," Tech. Rep. NCRG/94/004, Aston University, 1994.

[18] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social LSTM: Human trajectory prediction in crowded spaces," in *Proc. CVPR*, pp. 961–971, 2016.

[19] N. Lee *et al.*, "DESIRE: Distant future prediction in dynamic scenes with interacting agents," in *Proc. CVPR*, pp. 336–345, 2017.

[20] F. Li, C. He, X. Li, J. Peng, and K. Yang, "Geometric analysis-based 3D anti-block UAV deployment for mmWave communications," *IEEE Commun. Lett.*, vol. 26, no. 11, pp. 2799–2803, 2022.

[21] S. Parial, S. C. Ghosh, and A. K. Ghosh, "User-UAV association for dynamic user in mmWave communication for eMBB and URLLC," in *Proc. SoftCOM*, pp. 1–16, 2025.

[22] M. Bennis, M. Debbah, and H. V. Poor, "Ultrareliable and low-latency wireless communication: Tail, risk, and scale," *Proc. IEEE*, vol. 106, no. 10, pp. 1834–1853, 2018.

[23] T. S. Rappaport, S. Sun, R. Mayzus, H. Zhao, Y. Azar, K. Wang, G. N. Wong, J. K. Schulz, M. Samimi, and F. Gutierrez, "Millimeter wave mobile communications for 5G cellular: It will work!" *IEEE Access*, vol. 1, pp. 335–349, 2013.

[24] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*, Cambridge University Press, 2005.

[25] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, no. 3, pp. 379–423, 1948.

[26] T. C. Hesterberg, "Weighted average importance sampling and defensive mixture distributions," *Technometrics*, vol. 37, no. 2, pp. 185–194, 1995.

[27] A. B. Owen and Y. Zhou, "Safe and effective importance sampling," *J. Amer. Statist. Assoc.*, vol. 95, no. 449, pp. 135–143, 2000.

[28] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *Proc. ICML*, pp. 1861–1870, 2018.

[29] R. Jain, D.-M. Chiu, and W. R. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared computer systems," DEC Research Report TR-301, 1984.