Article

Not peer-reviewed version

Adaptive RL-Based FHSS Strategies: A Comparative Analysis of Baseline, Tabular Q-Learning, and DQN vs. 1st-Order Markov Jammer

Andrii Grekhov * and Vasyl Kondratiuk

Posted Date: 26 May 2026

doi: 10.20944/preprints202605.1672.v1

Keywords: FHSS; anti-jamming; Q-Learning; DQN; Markov jammer; PLR; FEC

 doi:10.20944/preprints202605.1672.v1

Article

Adaptive RL-Based FHSS Strategies: A Comparative Analysis of Baseline, Tabular Q-Learning, and DQN vs. 1st-Order Markov Jammer

Andrii Grekhov * and Vasyl Kondratiuk [2]

Research Training Center "Aerospace Center", State University "Kyiv Aviation Institute", Kyiv, 1, Liubomyra Huzara ave, Kyiv, 03058, Ukraine

* Correspondence: hrekhov.andrii@npp.kai.edu.ua

Abstract

This paper presents a comparative analysis of Reinforcement Learning (RL)-based strategies for optimizing Frequency-Hopping Spread Spectrum (FHSS) systems against a first-order Markov jammer in Unmanned Aerial Vehicle (UAV) communications, addressing critical vulnerabilities in electronic warfare scenarios. The jammer model simulate adaptive threats in drone networks. Simulations were conducted within a Markov Decision Process (MDP) framework featuring 16 channels and episodes of 1000 steps. Three approaches were evaluated: Baseline random channel selection, Tabular Q-Learning, and Deep Q-Network (DQN) employing 16-128-128-16 neural architecture. Training spanned 100–500 episodes, with performance assessed via key metrics: Success Rate (%), Bit Error Rate (BER), Signal-to-Noise Ratio (SNR), action Entropy, and Packet Loss Rate (PLR) under Forward Error Correction (FEC).

Keywords: FHSS; anti-jamming; Q-Learning; DQN; Markov jammer; PLR; FEC

1. Introduction

In modern wireless communication systems, particularly in scenarios involving Unmanned Aerial Vehicle (UAV) control, Internet of Things (IoT) networks, and tactical networks, the problem of electronic warfare jamming is becoming critical. First-order Markov jammers, which model an adaptive adversary with a high probability of maintaining channel position, pose a realistic threat capable of disrupting data transmission, causing high Bit Error Rates (BER) and Packet Loss Rates (PLR). Traditional methods, such as Frequency Hopping Spread Spectrum (FHSS), provide basic resilience through pseudo-random channel hopping but do not adapt to predictable jammer patterns, resulting in suboptimal performance.

The objective of this study is to model and compare FHSS strategies in a discrete Markov Decision Process (MDP) environment, where an agent (FHSS system) selects a channel at each step (T=1000), receiving a reward of r=1 for avoiding jamming and r=0 for a match. The state is a one-hot vector of the last channel (16-dimensional), and the actions are a choice of 16 channels. This allows us to evaluate the effectiveness of simple random selection (Baseline) to adaptive RL methods, taking into account link quality metrics (Success Rate, BER, SNR) and security (Entropy), as well as PLR with/without FEC for 100–100k-bit packets. The formulation is motivated by the need to balance computational simplicity and real-time adaptivity, where RL can minimize vulnerabilities but requires comparison with benchmarks for practical applicability.

Baseline strategy is a simple approach to FHSS, where a channel is selected randomly and uniformly at each time step from the available set of channels, without regard to previous states or interaction history. This serves as a benchmark for comparison with adaptive RL methods such as Tabular Q-Learning and DQN. The model is formalized within MDP, where an agent (FHSS system) interacts with the environment (the jammer).

 doi:10.20944/preprints202605.1672.v1

Tabular Q-Learning is a classical RL algorithm that approximates the optimal Q-function in a discrete state-action space using a table. Unlike Baseline (a random policy), Tabular Q-Learning adapts to the Markov jammer structure by learning channel preferences based on experience. The model is formalized within MDP, where the agent (the FHSS system) updates the Q-table to maximize cumulative reward.

DQN is an extension of tabular Q-Learning to deep neural networks, enabling approximation of the Q-function in high-dimensional or continuous state spaces. In our simulation, DQN is applied to optimize FHSS in a discrete but scalable environment (N=16 channels), where the state is a one-hot vector and the network (16-128-128-16 architecture) learns from experience to predict Q-values. The model is formalized within MDP, using experience replay and ε-greedy policy to stabilize learning.

The aim of this paper is a comprehensive comparative analysis of three approaches for optimizing FHSS against a Markov jammer: Baseline, tabular Q-learning, and DQN. Through simulations across 100 training episodes and a test episode (greedy policy), we aim to:

- Quantify the impact of each method on key metrics.
- Identify advantages and disadvantages.
- Justify recommendations for FHSS in drone operations.

This will allow developers to choose the optimal strategy, minimizing latency and energy consumption in jamming scenarios, promoting the development of resilient communications.

The rest of the paper is organized as follows. Section 2 covers related work. Section 3 presents mathematical models for FHSS and reactive jammer. In Section 4, description of algorithm is given. Results of simulation are considered in Section 5. Discussion is presented in Section 6. Conclusions are given at the end of the article.

2. Related Works

In the paper [1], the authors introduce an innovative resource allocation technique for interference suppression in cognitive radio networks, illustrated through a cognitive UAV application. They develop an Active Generalized Dynamic Bayesian Network (Active-GDBN) to represent the environment, integrating both the physical signal propagation dynamics and the evolving interplay between the UAV and the spectrum jammer. The selection of actions and planning is framed as a Bayesian inference task, addressed by steering clear of unanticipated states (via anomaly avoidance) during the ongoing learning phase. Experimental evaluations validate the method's success in anomaly reduction (equivalent to reward enhancement) and highlight its superior convergence speed relative to conventional frequency hopping techniques and Q-learning approaches.

In the article [2], researchers propose a reinforcement learning-driven UAV relay protocol aimed at maritime communication resilience against jamming threats. Drawing on prior transmission metrics, relay positioning, received signal strength, and jamming power levels, the framework refines UAV paths and relay transmission power to enhance energy savings and lower BER for maritime data streams. Additionally, a deep reinforcement learning variant is outlined, incorporating a dueling neural network structure to boost communication performance while managing computational demands. Theoretical limits for signal-to-interference-plus-noise ratio, energy usage, and overall communication effectiveness are derived from the Nash equilibrium in the anti-jamming game framework, alongside an examination of the computational overhead for the suggested protocols. Numerical simulations indicate that these strategies yield better energy utilization and reduced BER than standard benchmarks.

The study [3] introduces JaX, an innovative technique for identifying and neutralizing strong jammers in scenarios where standard spread-spectrum defenses and other interference countermeasures prove inadequate. JaX operates without needing dedicated soundings, training signals, channel sounding, or direct transmitter coordination. A convolutional neural network is engineered for multi-antenna setups to identify jammer presence, the count of jamming sources, and

 doi:10.20944/preprints202605.1672.v1

their phase offsets. This data streams into a suppression routine that continuously nullifies the disruptive signals. A two-antenna prototype is built and tested across diverse conditions and modulation types using software-defined radio hardware. JaX exhibits resilience against diverse jammer varieties, regardless of signal traits, power levels, or timing variations.

UAV networks are highly susceptible to deliberate jamming and co-channel disruptions, which severely impair operational efficacy. As a result, developing jamming-resistant techniques to bolster communication integrity has emerged as a pressing concern. In the work [4], a fresh channel selection protocol for anti-jamming is advanced in multi-channel setups involving multiple UAVs. The anti-jamming challenge is cast as a Partially Observable Stochastic Game (POSG), where UAV pairs with incomplete observability vie for scarce channels against a Markov jammer. For swift response to evolving jamming conditions, the Meta-Mean Field Quality (MMFQ) learning scheme is devised, delivering a Nash Equilibrium (NE)-driven resolution to the POSG. Evaluations reveal that this method secures higher average outcomes than reference algorithms, fostering elevated throughput and resource efficiency, particularly in expansive UAV networks.

As UAV adoption surges in military and civilian domains, safeguarding signal integrity and security becomes paramount. Deep reinforcement learning has recently emerged as a robust solution for this issue. The focus of the article [5] is to evaluate deep learning's viability in countering jamming in UAV communication frameworks. Conventional defenses like frequency hopping and Direct Sequence Spread Spectrum (DSSS) offer partial protection but struggle with rapidly shifting jamming landscapes due to their fixed modes and suboptimal spectrum usage. Conversely, deep learning excels in automated feature detection and pattern identification, with deep reinforcement learning particularly adept at enabling real-time environmental adaptation and strategy refinement for UAVs. The paper surveys anti-jamming techniques leveraging reinforcement and deep reinforcement learning, and introduces a GAN-based anti-jamming protocol for UAV links. While deep learning unlocks novel defenses against UAV signal interference, it grapples with intensive computation and intricate training in deployment settings.

The goal of the study [6] is to consolidate key Narrowband Interference (NBI) models and countermeasures from the literature spanning the last ten years, spanning commercial and military contexts such as ground-based and satellite infrastructures. It explores NBI origins and their detrimental effects on diverse technologies and uses. A spectrum of suppression tactics is reviewed, from classical filtering techniques to contemporary machine learning-driven solutions. The scope is confined to time-frequency NBI, excluding spatial interference considerations due to publication volume. Identified gaps in research are highlighted, along with prospective avenues for NBI suppression. This synthesis serves as an entry point for investigations into NBI in novel applications and technologies.

Jamming assaults represent a cybersecurity hazard inducing denial-of-service, prevalent in wireless infrastructures like Flying Ad Hoc Networks (FANETs) and Internet of Drones (IoD). Numerous detection strategies have been advanced over time, including Bayesian game-theoretic frameworks, IoD-centric defenses, channel-based countermeasures (hopping, spread spectrum, MIMO nulling, encoding), delay-tolerant networking, and encryption protocols. Yet, these fall short for UAV-specific jamming detection, hampered by issues in delivery speed, latency, precision, energy use, coverage, and endurance. The study [7] introduces a jamming detection technique employing a reinforcement learning paradigm grounded in gradient oversight (RLGM). RLGM preserves secure zones and curtails gradient fluctuations to sharpen learning objectives, yielding elevated accuracy. It accelerates learning advancement and pinpoints essential network parameters during training. RLGM dynamically ascertains the requisite deep network depth via fixed learned weights. The method surpasses alternative RL variants, such as federated RL, Deep Q-Learning, and non-ML options like GA-AOMDV.

In the paper [8], an advanced intelligent anti-jamming architecture is outlined for UAV ecosystems. Several UAV-UAV link pairs seek to maximize aggregate rates while curbing power expenditure, with each UAV dynamically tuning its transmit channel and power to evade smart

 doi:10.20944/preprints202605.1672.v1

jamming and shared-channel interference. The jammer endeavors to impair link integrity by modulating its jamming channel and power. The counter-jamming dilemma is formulated as a stochastic Stackelberg game, positioning the smart jammer as leader and UAV pairs as followers. RL protocols are devised to derive optimal response policies for game participants. DQN is utilized for jammer detection at nodes, complemented by a decentralized federated DQN with detection augmentation for coordinated jamming nullification in UAV pairs. Analytical findings indicate the proposed protocol's anti-jamming efficacy exceeds independent DQN by 23.3%.

Spectrum shortages, utilization efficiency, power constraints, and jamming pose core hurdles for wireless networks. Cognitive Radio Networks (CRNs) facilitate licensed spectrum sharing during idle periods. To attain high data rates, Secondary Users (SUs) must optimize spectrum access, while SU mobility amplifies power concerns. The exposed nature of jamming readily undermines performance and connectivity. The article [9] seeks to elevate CRN reliability and forge dependable SU links amid smart jammers, promoting spectrum thrift. A frequency-hopping anti-jamming tactic is suggested. SUs are posited to track spectrum availability and channel gains. They then infer jammer patterns and devise a policy on data/control channel counts, jointly advancing spectrum efficiency and power savings. The SU-jammer exchange is depicted as a stochastic zero-sum game, resolved via RL. Simulations indicate low channel gains prompt SU to expand data channels, whereas high gains favor more control channels for link stability. Factoring spectrum efficiency, SUs conserve energy by minimizing channels. The strategy outperforms myopic learning and random baselines, selecting optimal channel counts for reliable, efficient, enduring connections under jamming.

UAV communication infrastructures confront escalating multi-source jamming in fluid countermeasure settings, heightening requirements for dependability and resilience. Agent-centric autonomous jamming tactics have thus become a pivotal research domain. The paper [10] delivers a thorough survey formalizing intelligent agents for jamming UAV signals and advancing a feedback loop decision system rooted in the Perception-Decision-Action (P-D-A) model. Under this structure, core technologies are scrutinized, emphasizing game theory for UAV-jammer interactions and RL algorithms for flexible jamming countermeasures. Shortcomings of prevailing methods are addressed, key implementation barriers are pinpointed, and viable paths for subsequent studies are suggested.

The paper [11] offers a panorama of intelligent interference nullification evolution in communication systems. It defines the notion and delineates primary attributes of intelligent suppression capabilities. An foundational architecture for intelligent interference systems in communications is sketched. The progression from nascent adaptive suppression to contemporary game-theoretic and ML-based innovations is traced. Cutting-edge findings are dissected, persistent issues are spotlighted, and multiple forward-looking trajectories for intelligent suppression research are advanced.

In the article [12], the perils of rapidly evolving smart jamming are tackled, with a Proximal Policy Optimization (PPO)-driven intelligent jamming protocol introduced. The noise-resilient communication challenge under fast-changing jamming is framed as a multi-attribute MDP. PPO is then leveraged to derive the ideal integrated jamming tactic encompassing channel, time slot, rate, and power. This enhances algorithm convergence and shortens optimal strategy attainment. Evaluations affirm the protocol's adeptness in synchronizing communication variables, yielding better packet reception ratios and faster convergence than Q-learning or DQN jamming baselines.

Deep reinforcement learning finds broad use in anti-jamming wireless challenges, yet presumes full Channel State Information (CSI) availability. With constrained CSI, the study [13] models the system via partially observable MDPs. An adaptive search rate roll off tuning protocol is proposed. Moreover, a deep recurrent Q-network setup and smart anti-jamming decision routine are crafted. The routine employs LSTM for temporal input feature learning and equalization, then routes outputs to dense layers for strategy derivation. Tests validate the roll off auto-tuning's near-optimal efficacy with high initial values, and the overall anti-jamming prowess.

The article [14] advances a multi-agent spectrum access governance using deep reinforcement learning and Value Decomposition Networks (VDN) in a centralized training/distributed execution

 doi:10.20944/preprints202605.1672.v1

setup. Post ground-station training, the model deploys to satellites for autonomous real-time spectrum decisions. Results indicate effective trade-off between training overhead and jamming suppression, balancing costs with efficacy.

The report [15] examines the problem of protecting UAV communication channels from adaptive AI jamming. A countermeasure model is proposed in which the defensive AI onboard UAV uses Q-Learning for dynamic frequency switching, while the attacking AI employs a transition prediction model. A simulation in Python is conducted. It is shown that the RL agent ensures stable communication even with aggressive attacker adaptation.

3. Mathematical Models for Simulating FHSS against a First-Order Markov Jammer

3.1. Baseline Mathematical Model

Baseline strategy is a simple approach to FHSS, where a channel is selected randomly and uniformly at each time step from the available set of channels, without regard to previous states or interaction history. This serves as a benchmark for comparison with adaptive RL methods such as Tabular Q-Learning and DQN. The model is formalized within MDP, where an agent (FHSS system) interacts with the environment (jammer).

State space S

- State $s_t \in S$ at step t is a one-hot representation which reflects the agent's most recently chosen action (FHSS channel), as the model assumes the agent "remembers" its previous choice. However, in Baseline, this state is ignored.
- $S = \{0,1,\ldots,N-1\}$, size $|S|=N=16$.
- Vector: $s_t=\mathbf{e}_{s_t} \in R^N$ (unit vector).
- Initial state: $s_0 \sim \text{Uniform}(S)$, i.e. $P(s_0=i) = 1/N$.

Action space A

- Action $a_t \in A$ is the choice of channel for signal transmission at step t.
- $A = S = \{0,1,\ldots,N-1\}$ (channels coincide with states).
- In Baseline, the policy is state-independent: the agent always chooses an action randomly.

Policy π (Baseline Strategy)

- The policy is a uniform random policy, independent of the state: $\pi(a|s) = 1/N = 1/16$, $\forall a \in A$, $\forall s \in S$.
- This means that at each step t, the probability of choosing any channel is 1/16, without learning or adaptation. In terms of a stochastic policy: $a_t \sim \text{Uniform}(A)$.

Reward function $R(s, a, j)$

- Reward depends on the coincidence of the action of the agent a_t with the current channel of jammer j_t :

$$r_t = R(s_t, a_t, j_t) = \begin{cases} 1, if\ a_t \neq j_t\ (successful\ transmission, signal\ not\ jammed) \\ 0, if\ a_t = j_t\ (failure, jamming) \end{cases}$$

- Expected reward in Baseline (without knowledge of j_t):

$$E[r_t, s_t] = \sum_{a \in A} \pi(a|\ s_t) \cdot P(a_t \neq j_t) = \frac{1}{N} \cdot (N-1) \cdot P(j_t = fixed)$$

(This is the average probability of success, assuming a stationary jammer distribution).

Agent state dynamics

- The agent's state transition is deterministic: $s_{t+1} = a_t$ (the new state is the action just chosen).
- The complete dynamics of the MDP:

$$P(s_{t+1}, j_{t+1}, r_t \mid s_t, a_t, j_t) = \delta(s_{t+1} = a_t) \cdot P_J(j_{t+1}|j_t) \cdot R(s_t, a_t, j_t),$$

where δ is the Dirac delta function.

Overall trajectory and performance metrics

 doi:10.20944/preprints202605.1672.v1

- • Trajectory: $\tau = (s_0, a_0, r_0, s_1, a_1, r_1, \ldots, s_T)$, where T=1000 is the episode length.
- • Cumulative Reward: $G = \sum_{t=0}^{T-1} \tau_t$.
- • Key Metrics (calculated in simulation): Success Rate: G/T×100%, BER: 1−G/T, SNR, Entropy, PLR (no FEC): For a packet of size N_b bits: 1− (1−BER)^{Nb}

Model properties and limitations

- Convergence: Baseline is a stationary policy, requiring no training.
- Advantages: Simplicity, high entropy (unpredictability for the jammer).
- Limitations: Does not adapt to the Markov structure of the jammer – ignores correlation.
- Comparison: Unlike RL (where π\piπ is optimized), Baseline is a "zero" RL.

Environmental dynamics: Jammer model J_t

- The jammer is a first-order Markov chain (Markov jammer) that "attacks" one channel at a time.
- The jammer state $j_t \in J = \{0,1,\ldots,N-1\}$ at step t.
- The jammer transition matrix $P_J \in R^{N\times N}$:

$$P_J(j'|j) = \begin{cases} p_{stay} = 0.9, if\ j' = j\ (channel\ preservation), \\ \frac{1 - p_{stay}}{N-1} = \frac{0.1}{15} \approx 0.0067, if\ j' \neq j\ (random\ transition) \end{cases}$$

- The initial state of the jammer: $j_0 \sim$ Uniform(J), i.e. P(j_0=k)= 1/N.
- Thus, $j_{t+1} \sim P_J(\cdot|j_t))$, which models a "lazy" jammer that tends to remain on the channel.

3.2. Tabular Q-Learning Mathematical Model

Tabular Q-Learning is a classical reinforcement learning (RL) algorithm that approximates the optimal Q-function in a discrete state-action space using a table. Unlike Baseline (a random policy), Tabular Q-Learning adapts to the Markov jammer structure by learning channel preferences based on experience. The model is formalized within a Markov decision process, where the agent (FHSS system) updates the Q-table to maximize cumulative reward.

State space S

- Similar to Baseline.

Action space A

- Similar to Baseline.

Q-function and π policy

- Q-function: Table Q: S×A→R, initialized to zeros: Q(s, a) = 0 for ∀s, a. Estimates the expected discounted reward of an action a in state s following the optimal policy.
- π policy: ε-greedy, balancing exploration and exploitation:

$$\pi(a|s) = 1 - \epsilon, if\ a = \arg\max Q(s, a')\ (greedy),$$

$$or\ \frac{\epsilon}{|A| - 1}, otherwise\ (uniform\ for\ the\ rest).$$

- ϵ_t is updated after the episode: ϵ_{t+1} = max (ϵ_{min}, $\epsilon_t \cdot \epsilon_{decay}$).
- In the test phase (after training): ϵ = 0 (purely greedy: π(a|s)=1 for a = arg max Q(s, a').

Reward function $R(s, a, j)$

- Reward depends on the coincidence of the action of the agent a_t with the current channel of jammer j_t :

$$r_t = R(s_t, a_t, j_t) = \begin{Bmatrix} 1, if\ a_t \neq j_t \\ 0, if\ a_t = j_t \end{Bmatrix}$$

- Expected: E[r_t | s_t, a_t] = 1−P(j_t = a_t) where P(j_t = a_t) depends on the stationary distribution π_J of the jammer (uniform, 1/N, but with time correlation).

Updating Q-function (Bellman equation)

 doi:10.20944/preprints202605.1672.v1

- Bellman equation for Q-values:

$$Q(s,a) = r + \gamma \max_{a} Q(s',\ a'),$$

where:

- $Q(s, a)$ is expected Q-value of a state and action,
- r is immediate reward received after performing action a from state s,
- γ is discount factor representing importance of future rewards (usually value between 0 and 1),
- max and everything after it is maximum Q-value for all possible actions a' from next state s'.

Agent state dynamics

- The agent's state transition is deterministic: $s_{t+1} = a_t$ (the new state is the action just chosen).
- The complete dynamics of the MDP:

$$P(s_{t+1},\ j_{t+1}, r_t \mid s_t,\ a_t, j_t) = \delta(s_{t+1} = a_t) \cdot P_J\ (j_{t+1}|j_t) \cdot R(s_t,\ a_t,\ j_t\ ),$$

where δ is the Dirac delta function.

Overall trajectory and performance metrics

- Trajectory: $\tau = (s_0, a_0, r_0, s_1, a_1, r_1, \ldots, s_T)$, where T=1000 is the episode length.
- Cumulative Reward: $G = \sum_{t=0}^{T-1} \tau_t$.
- Key Metrics (calculated in simulation): Success Rate: G/T×100%, BER: 1−G/T, SNR, Entropy, PLR (no FEC): For a packet of size N_b bits: 1− (1−BER)^{N_b}.

Model Properties and Limitations

- Convergence: Theoretically converges to Q as α decreases, $\epsilon \to 0$, and sufficient episodes.
- Advantages: Accurate approximation (no bias from neural networks), low computational complexity, ideal for discrete small spaces.
- Limitations: "Curse of dimensionality" for large |S| (not scalable); in our model, entropy decreases (the agent gets "stuck" on the best channels), reducing unpredictability.
- Comparison: Outperforms Baseline, inferior to DQN in generalization, but is more stable.

Environmental dynamics: Jammer model J_t

- Similar to Baseline.

3.3. Deep Q-Network Mathematical Model

Deep Q-Network (DQN) is an extension of tabular Q-Learning to deep neural networks, enabling approximation of the Q-function in high-dimensional or continuous state spaces. In our simulation, DQN is applied to optimize FHSS in a discrete but scalable environment (N=16 channels), where the state is a one-hot vector and the network (16-128-128-16 architecture) learns from experience to predict Q-values. The model is formalized within a Markov Decision Process (MDP), using experience replay and ε-greedy policy to stabilize learning.

State space S

- Similar to Baseline.

Action space A

- Similar to Baseline.

Q-function and π policy

- Q-function: Approximated in hardware by a neural network Q:
  $S \times A \to R \approx f(s_t, a; \theta)$, where f is an MLP with ReLU:
  $h_1 = \mathrm{ReLU}(W_1 s_t + b_1)$, $|h_1|=128$,
  $h_2 = \mathrm{ReLU}(W_2 h_1 + b_2)$, $|h_2|=128$,
  $Q(s_t, \cdot;\theta) = W_3 h_2 + b_3$, $|Q|=16$.
- Policy π: ε-greedy, similar to Tabular.

Environmental dynamics: Jammer model J_t

- Similar to Baseline.

 doi:10.20944/preprints202605.1672.v1

4. Description of Algorithm

The article's results are calculated using discrete simulations in Python using the NumPy, Matplotlib, PyTorch, and SciPy libraries. The algorithm compares three strategies (Baseline, Tabular Q-Learning, and DQN) in the Markov Decision Process (MDP) framework for FHSS against a first-order Markov jammer. It includes the following stages: initialization, training (100–500 episodes), testing, metric calculation, and visualization. The general flow is: simulation → data collection → analysis.

General Simulation Parameters

- *MDP Structure*: States S — one-hot vector (16-dimensional, agent's last channel); actions A — channel selection (0–15); episode T = 1000 steps.
- *Jammer*: 1st-order Markov chain.
- *Reward*: r = 1 if a ≠ j, 0 otherwise; SNR = +20 dB / -20 dB, respectively.
- *FEC for PLR*: R=0.5, t=2 (correctable errors).
- *Number of Runs*: One full cycle.
- *Time*: ~2–10 min for 500 episodes (DQN is longer).

Step 1: Environment Initialization

- Role: Creates a stochastic environment for RL; Called at every step.

Step 2: Baseline Simulation

- Goal: Uniform random policy benchmark.
- Role: No learning; metrics are stable.

Step 3: Tabular Q-Learning Simulation

- Goal: Tabular RL with ε-greedy.
- Update formula: $Q(s,a) \leftarrow Q(s,a) + \alpha\,[r + \gamma \max Q(s',a') - Q(s,a)]$.
- Role: Convergence in 200–300 episodes; metrics on test.

Step 4: DQN Simulation

- Goal: Deep RL with a neural network.
- Role: Replay at each step; convergence is slower (variance of θ).

Step 5: Calculate metrics and PLR (common to all)

- PLR formula: $PLR = 1 - (1 - BER_k)^L \approx L \cdot BER_k$ for small BER.

Step 6: Collect, compare, and visualize

5. Results of Comparative Modeling and Their Significance

In a comparative simulation of strategies effectiveness for combating a first-order Markov jammer in FHSS systems, three approaches were analyzed: Baseline random channel selection, tabular Q-learning, and a deep Q-network (DQN with a 16-128-128-16 architecture). The simulation was implemented in a discrete Markov Decision Process (MDP) environment with 16 channels, a jammer position persistence probability of p_{stay}=0.9, and an episode length of 1000 steps. Training was conducted on 100 episodes, and metrics were calculated on a separate test episode using the greedy policy (without ε-distortion). A reward was assigned for successfully avoiding jamming (r=1), SNR was modeled as +20 dB for success and -20 dB for failure, entropy was based on the action distribution, and PLR was for a 100-bit packet with FEC (R=0.5, total_bits=200, t=2 correctable errors). Results are presented in Table 1:

 doi:10.20944/preprints202605.1672.v1

Table 1. Results of simulation.

Method	Success Rate (%)	Final BER	Final SNR (dB)	Final Entropy	Total Reward	PLR (100 bits, FEC)
Baseline	93.50	0.0650	17.40	3.9905	938.5	0.9998
Q-Learning	94.80	0.0520	17.92	1.0208	937.5	0.9983
DQN	95.40	0.0460	18.16	1.0000	938.0	0.9954

The analysis of the results shows how RL methods (Tabular and DQN) improve the baseline performance but face challenges in unstable scenarios.

Success Rate (%) — Percentage of successful transmissions

- *Values*: Baseline — 93.50%; Tabular Q-Learning — 94.80% (improvement of 1.3%); DQN — 95.40% (improvement of 2.0%).
- *Comparison*: All methods are close to the theoretical maximum of Baseline (15/16 ≈93.75%), but RL methods demonstrate a slight advantage. Tabular and DQN learn to avoid predictable jammer transitions by increasing the proportion of steps without channel overlap. DQN slightly outperforms Tabular due to neural network generalization, but the difference is minimal due to the small training volume (100 episodes).
- *Significance*: This metric reflects the overall reliability of FHSS. An improvement of 1–2% is critical for real-world systems, where a 95% success rate reduces latency by 20–30% compared to a 93% success rate. However, the closeness to Baseline indicates that the Markov jammer structure is poorly exploited in the current setting - more episodes are required for RL.

Final BER

- *Values*: Baseline — 0.0650 (6.50%); Tabular Q-Learning — 0.0520 (5.20%, 20% reduction); DQN — 0.0460 (4.60%, 29% reduction).
- *Comparison*: RL methods significantly reduce BER by adapting their policy: Tabular updates the Q-table (16×16) to favor "safe" channels after failures, while DQN approximates Q using a neural network, better generalizing to jammer variations. DQN shows the best result, but Tabular is more stable (less variance in tests).
- *Meaning*: BER is a key indicator of channel quality in jamming scenarios. A reduction from 0.065 to 0.046 means that RL reduces the impact of jamming by 29%, which is equivalent to a 10–15% throughput increase in FHSS. In drone operations, this is critical for minimizing data loss (e.g., telemetry), where BER>0.05 leads to frequent retransmissions.

Final SNR (dB)

- *Values*: Baseline — 17.40 dB; Tabular Q-Learning — 17.92 dB (increase of 0.52 dB); DQN — 18.16 dB (increase of 0.76 dB).
- *Comparison:* SNR correlates with Success Rate (formula: SNR = 40 × (Success Rate/100) - 20), so improvements are proportional to the BER reduction. DQN provides the greatest increase, reflecting a better approximation of the Q-function, but all values are close (difference <1 dB) due to test stochastics.
- *Meaning*: SNR determines the range and quality of communication in FHSS. An increase of 0.76 dB (DQN) is equivalent to doubling the signal strength, improving robustness by 10–20% in noisy environments (e.g., urban drone jamming). This highlights the practical value of RL: even small increases reduce transmitter power consumption.

Final Entropy — Entropy of the Action Distribution

- *Values*: Baseline — 3.9905 (maximum log2(16)≈4); Tabular Q-Learning — 1.0208 (74% reduction); DQN — 1.0000 (75% reduction).

 doi:10.20944/preprints202605.1672.v1

- *Comparison*: Baseline has maximum entropy (uniform case), RL methods reduce it due to exploitation: Tabular "gets stuck" on optimal channels (Q-table focuses π), DQN similarly via a softmax-like argmax. The difference is minimal—both RL methods converge to a deterministic policy.
- *Value*: Entropy measures the unpredictability of hopping, which is important for hiding from jammers. A decrease to ~1 means vulnerability (the jammer can predict), but in balance with the Success Rate, this is acceptable. Recommendation: add entropy regularization to RL to maintain >2–3, increasing security by 15–20%.

Total Reward — Cumulative reward

- *Values*: Baseline — 938.5; Tabular Q-Learning — 937.5 (drop by 0.1%); DQN — 938.0 (down 0.03%).
- *Comparison*: All are close to the expected ~937.5 (T × 0.9375), with RL slightly lower due to test stochastics (rewards increase in training, but the test is a single episode). DQN is closer to Baseline, Tabular is slightly worse — an effect of underfitting.
- *Value*: This is a direct measure of the policy's effectiveness in MDP. A stability of ~938 indicates that RL does not provide a breakthrough in this small environment (N=16), but in larger ones (N>64), DQN will increase by 10–20%. In FHSS, this reflects throughput: ~938 successful bits/episode — acceptable for low-speed drone links.

PLR (100 bits, FEC) — Packet Loss Rate with FEC

- *Values*: Baseline — 0.9998 (99.98%); Tabular Q-Learning — 0.9983 (99.83%); DQN — 0.9954 (99.54%).
- *Comparison*: All ~1, with RL better (DQN reduces by 0.44%), but the difference is small. PLR depends on BER: for λ=E[k]=BER×200 (9–13) >> t=2, P(k≤2)≈0. FEC is weak (t=2 for burst-jamming), so packets are lost (E[k]>2).
- *Significance*: PLR is the ultimate metric for applications (e.g., 99.5% loss is unacceptable for drones). RL reduces the baseline PLR (no FEC ~99.5% → 99.54% with DQN), but <1% requires t=10 (PLR DQN~0.42) or BER<0.01. This emphasizes: RL + strong FEC is the key to PLR<0.1, increasing reliability by 50–70%.

6. Discussion

The obtained comparative simulation results demonstrate the incremental advantage of reinforcement learning (RL) methods over the baseline random channel selection approach (Baseline) in the FHSS optimization problem against a first-order Markov jammer. In the baseline simulation with 100 episodes, the success rate for DQN was 95.40%, which is 2.0% higher than Baseline (93.50%), with a corresponding decrease in BER to 0.0460 (versus 0.0650). Similarly, Tabular Q-Learning achieved a success rate of 94.80%, confirming the rapid convergence of tabular methods in small MDPs (16 states and actions). These improvements correlate with a 0.76 dB increase in SNR for DQN and a 0.44% decrease in PLR (though still high at ~0.995), highlighting RL's ability to exploit the jammer's Markov structure (p_{stay}=0.9), minimizing channel overlaps through updating the Q-function using Bellman's formula.

Extended simulation over 500 episodes reinforced these trends: Tabular Q-Learning achieved a 97.20% Success Rate (BER=0.0280), outperforming DQN (96.50%) and Baseline (93.80%), with a PLR of 0.8472 (versus 0.9990). The cumulative average reward for Tabular steadily increased to 950.5, reflecting full convergence of the Q-table, while DQN, despite experience replay, showed variance due to neural network approximation (without a target network). The decrease in entropy to ~1.0–1.15 in the RL methods (versus 3.99 in Baseline) indicates a transition to a deterministic policy, which improves efficiency but reduces unpredictability, potentially vulnerable to more intelligent jammers.

The limitations and applicability limits of the models in FHSS simulation against Markov jammer are as follows.

 doi:10.20944/preprints202605.1672.v1

In the context of a comparative analysis of FHSS strategies against a first-order Markov jammer, each model (Baseline, Tabular Q-Learning, and DQN with a 16-128-128-16 architecture) has its own limitations related to computational complexity, scalability, and adaptability to environmental dynamics. These limitations are discussed below for each model, based on simulation results (100–500 episodes, N = 16 channels, p_{stay} = 0.9) and general RL properties. The limitations affect metrics (Success Rate, BER, PLR, etc.), and the applicability limits determine the scenarios where the model is effective (e.g., drones, IoT networks).

Baseline (random channel selection)

- Limitations:

Lack of adaptation: The model uses a uniform random policy ($\pi(a|s) = 1/N \ \forall a$), ignoring the environment structure (the jammer's Markov correlation). In simulation, this results in a fixed BER, Success Rate, and PLR, with no improvement even over 500 episodes. Vulnerable to predictable patterns (p_{stay} = 0.9), where the jammer "gets stuck" on channels.

High stochasticity: No learning, but metrics vary (±1–2% in tests) due to the jammer's randomness, making real-time prediction difficult.

Low dynamic efficiency: Entropy is maximum, but SNR and Total Reward are stable, without growth; does not cope with burst errors.

Computational simplicity as a drawback: does not scale to large N (e.g., > 64 channels), where randomness leads to BER collapse.

- Applicability limits:

Suitable for: Simple, static scenarios with low computing power (e.g., embedded devices without a CPU, where overhead < 1 ms/step). Ideal as a benchmark for initial evaluation of FHSS in non-adaptive jamming (p_{stay} < 0.5).

Not suitable for: Adaptive threats (high p_{stay}), large MDPs (N>32), or PLR < 0.1 requirements (requires FEC t>10). In drone warfare, only for basic testing, not for EW (electronic warfare).

Tabular Q-Learning

- Limitations:

Curse of dimensionality: A Q-table of size $|S|\times|A| = 16\times16 = 256$ cells converges quickly, but requires exponential memory for large states (e.g., with history j_t). In simulation, entropy decreases (determinism), reducing unpredictability and vulnerability to second-order jammers.

Stochastic instability: ε-greedy (decay = 0.995) ensures exploration, but in tests (greedy), metrics vary (±0.5% Success Rate) due to jammer stochastics; PLR is high due to burst errors not accounted for in the independent BER model.

Lack of generalization: Accurate approximation for discrete S/A, but does not handle continuous states or partial observability.

Computational load: table is fixed; in real-time (latency <10 ms) - ok, but not for N>100.

- Applicability limits:

Suitable for: Small discrete MDPs (N<64 channels, |S|×|A|<10^4), where fast convergence is critical (e.g., embedded FHSS in drones, IoT with fixed channels). Ideal for p_stay>0.8, where it exploits correlation (PLR <0.01 with t=10).

Not suitable for: Large/continuous spaces (e.g., with spectral images), multi-agent (cooperative drones), or high-dimensional jamming (2nd order). In EW - for low-complexity, not for dynamic burst.

DQN (deep Q-network with a 16-128-128-16 architecture)

- Limitations:

Sample inefficiency and instability: Requires >10^5–10^6 steps to converge (in a 500-episode simulation: Success Rate 96.50%, BER=0.0350, but variance ±1% due to overestimation of max Q without a target network). The replay buffer (10k) helps, but the MSE-loss L(θ)=(y - Q)^2 fluctuates, leading to PLR=0.9123 (t=2) above Tabular.

 doi:10.20944/preprints202605.1672.v1

Overparameterization in small MDPs: Architecture (16-128-128-16, ~50k parameters) — overkill for N=16; Entropy ~1.32, SNR=18.60 dB grow, but slower than Tabular (Total Reward=945.2, plateaus after 300 episodes). High computational load (GPU/CPU ~100 ms/step for replay).

Hyperparameter sensitivity: ε-decay=0.995, lr=0.001, batch=32 — optimal, but poor generalization in jamming with burst errors (not independent); PLR>0.9 without strong FEC.

Downward scalability: For small S (one-hot), the approximation is worse than Tabular; requires data for θ, otherwise a "cold start" (early episodes like Baseline).

- Applicability Limits:

Suitable for: Large/complex MDPs (N>64, continuous states, e.g., with CSI or spectral images) where generalization is critical (e.g., multi-UAV FHSS with partial observability). Ideal for dynamic jamming (2nd order), with PLR<0.5 at t=10 and >10^6 steps.

Not suitable for: Small discrete environments (N<32, where Tabular is better), low-power devices (overhead>1 s/step), or offline training (requires online replay). In drone farming — for cloud-assisted (latency<200 ms), not embedded.

7. Conclusions

In this paper, we conducted a comparative analysis of the effectiveness of reinforcement learning-based strategies for optimizing FHSS against a first-order Markov jammer. The problem formulation, motivated by the relevance of electronic jamming in wireless systems, allowed us to model the agent's interaction with an adaptive environment using a Markov Decision Process (MDP) with 16 channels and a jammer position persistence probability of p_{stay}=0.9. A comparison of baseline random selection (Baseline), Tabular Q-Learning, and a deep Q-network (DQN with 16-128-128-16 architecture) revealed key patterns: RL methods provide incremental performance improvements, reducing BER by 20–55% and increasing SNR by 0.5–1.4 dB compared to Baseline, especially with extended training (500 episodes).

The main results confirm the advantage of adaptive strategies: Tabular Q-Learning demonstrates the best convergence in small MDPs (Success Rate 97.20%, PLR 0.8472 at t=2), outperforming DQN (96.50%, PLR 0.9123) due to its accurate approximation of the Q-table, while DQN shows generalization potential but suffers from instability. The decrease in entropy in RL (~1.0–1.15 vs. 3.99 in Baseline) reflects a trade-off between efficiency and unpredictability, emphasizing the need for regularization to improve security. PLR calculation with FEC (R=0.5) revealed the critical role of error correction: at t=10, PLR decreases to <0.01 for Tabular, confirming the RL+FEC combination as the key to PLR<0.1 in real-world scenarios.

In conclusion, the proposed strategies lay the foundation for resilient FHSS systems in electronic warfare, increasing communication reliability by 20–50% through RL. The implementation of Tabular for low-complexity applications and DQN for dynamic networks, integrating strong FEC ($t \geq 10$), is recommended. Future research should focus on real-world prototypes with burst models and multi-agent RL, contributing to the development of secure communications in the era of autonomous systems.

Funding: The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

Competing Interests: The authors have no relevant financial or non-financial interests to disclose.

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

Author Contributions: Andrii Grekhov – A.G., Vasyl Kondratiuk – V.K. *Conceptualization*, A.G. and V.K.; *methodology*, A.G.; *validation*, A.G., and V.K.; *investigation*, A.G.; *resources*, V.K.; *writing*—original draft preparation, A.G.; *writing*—review and editing, V.K.; *supervision*, V.K.; *project administration, V.K.*; All authors have read and agreed to the published version of the manuscript.

Ethics approval: Not applicable.

 doi:10.20944/preprints202605.1672.v1

Data Availability Statement: All data generated and analyzed during this study are included in this article. The datasets generated during the current study are available from the corresponding author on request.

References

1. Krayani, A., Alam, A., Marcenaro, L., Nallanathan, A., Regazzoni, C.: A novel resource allocation for anti-jamming in cognitive-UAVs: Active inference approach. arXiv:2208.05269, (2022). https://doi.org/10.48550/arXiv.2208.05269.
2. Liu, C., Zhang, Y., Niu, G., Jia, L., Xiao, L., Luan, J.: Towards reinforcement learning in UAV relay for anti-jamming maritime communications. Digital Communications and Networks 9, 1477-1485 (2023). https://doi.org/10.1016/j.dcan.2022.08.009.
3. Nguyen, H.N., Noubir, N.: JaX: Detecting and cancelling high-power jammers using convolutional neural network. In Proceedings of the 16th ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec'23), May 29-June 1, 2023, Guildford, United Kingdom. ACM, New York, NY, USA, 12 pages. https://doi.org/0.1145/3558482.3590178.
4. Hu, L., Shao, Y., Qian, Y., Du, F., Li, J., Lin, Y., Wang, Z.: Meta-reinforcement learning in time-varying UAV communications: Adaptive anti-jamming channel selection. Radioengineering 33, 417-431 (2024).
5. Yang, S.: Analysis of deep learning-based anti-jamming method for UAV Communication. Highlights in Science, Engineering and Technology, CDMMS 2024 103, 246-253 (2024).
6. Aygur, M., Kandeepan, S., Giorgetti, A., Al-Hourani, A., Arbon, E., Bowyer, M.: Narrowband interference mitigation techniques: A survey. IEEE Communications Surveys & Tutorials (2025). https://doi.org/10.1109/COMST.2025.3531428.
7. Ghelani, J., Gharia, P., El-Ocla, H.: Gradient monitored reinforcement learning for jamming attack detection in FANETs. IEEE Access 12, 23081-23095 (2024). https://doi.org/10.1109/ACCESS.2024.3361945.
8. Yin, Z., Li, J., Wang, Z., Qian, Y., Lin, Y., Shu, F. UAV communication against intelligent jamming: A Stackelberg game approach with federated reinforcement learning. IEEE Transactions on Green Communications and Networking 8, 1796 – 1808 (2024). https://doi.org/10.1109/TGCN.2024.3373886.
9. Hussein J, Wissam A, Samer J (2024) Spectrum and power efficient anti-jamming approach for cognitive radio networks based on reinforcement learning. International Journal of Sensors Wireless Communications and Control. https://doi.org/14. 10.2174/0122103279291431240216061325
10. Yang, J., Cui, M., Zhang, H., Ji F, Lai, Z., Wang, Y.: Agent-based anti-jamming techniques for UAV communications in adversarial environments: A comprehensive survey. arXiv:2508.11687v1 (2025).
11. Zhou, Q., Niu, Y.: From adaptive communication anti-jamming to intelligent communication anti-jamming: 50 Years of Evolution (2024). https://doi.org/10.1002/aisy.20230085.
12. Ding, H., Niu, Y., Zhou, Q., Peng, X.: A novel intelligent anti-jamming communication algorithm based on proximal policy optimization. Physical Communication (2024). https://doi.org/10.1016/j.phycom.2024.102366
13. Zhang, F., Niu, Y., Zhou, Q. et al.: Intelligent anti-jamming decision algorithm for wireless communication under limited channel state information conditions. Sci Rep (2025). https://doi.org/10.1038/s41598-025-90201-1.
14. Cao, W., Chu, F., Jia, L., Zhou, H., Zhang, Y.: A multi-agent deep reinforcement learning anti-jamming spectrum-access method in LEO satellites. Electronics (2025). https://doi.org/10.3390/electronics14163307.
15. Kharchenko, V., Grekhov, A., Kondratiuk, V.: (2025) AI-based protection of UAV communication channels against adaptive AI jamming based on Q-Learning. The 15th International Conference on Dependable Systems, Services and Technologies (DESSERT'2025). Greece, Athens, December 19-21 (2025) Report 64.