Received 3 July 2024; revised 5 August 2024; accepted 22 August 2024. Date of publication 28 August 2024; date of current version 5 September 2024.

Digital Object Identifier 10.1109/OJCOMS.2024.3451288

Leveraging Transformer Models for Anti-Jamming in Heavily Attacked UAV Environments

IBRAHIM ELLEUCH [1], ALI POURRANJBAR [1], AND GEORGES KADDOUM [1,2] (Senior Member, IEEE)

[1]Resilient Machine learning Institute, École de Technologie Supérieure, University of Quebec, Montreal, QC H7L 5R8, Canada

[2]Cyber Security Systems and Applied AI Research Center, Lebanese American University, Beirut 03797751, Lebanon

CORRESPONDING AUTHOR: I. ELLEUCH (e-mail: elleuch.ibrahim@gmail.com)

ABSTRACT In recent years, due to their ability to transmit and relay wireless signals in challenging terrains, Unmanned Aerial Vehicles (UAVs) and High Altitude Platform Stations (HAPS) have become indispensable in various operations in security, emergency, and military campaigns. However, these networks' ad-hoc structure and open nature make them highly vulnerable to numerous threats and, in particular, to severe jamming attacks. Furthermore, the communication link between a HAPS and multiple UAVs is also under the threat of multiple and different jamming attacks. Addressing these challenges requires innovative and novel methods capable of interactive and proactive defence strategies. To this end, in this study, we propose a method that combines a pseudo-random (PR) algorithm for initial channel selection with a Transformer-based module to predict jammer behavior. This proactive approach significantly enhances the robustness of UAV communications. Our results demonstrate substantial improvements in transmission success rates and prediction accuracy, offering a robust solution for secure UAV and HAPS communications under adverse conditions.

INDEX TERMS Anti-jamming, smart jamming, multiple-jamming, transformer, LSTM, UAVs, HAPS.

I. INTRODUCTION

UNMANNED Aerial Vehicles (UAVs) and High Altitude Platform Stations (HAPS) have emerged as pivotal components in today's rapidly evolving technological landscape. Their applications are diverse and far-reaching, ranging from entertainment to critical military operations, which highlights their versatility and indispensability. As airborne devices capable of transmitting and relaying wireless signals, UAVs and HAPS have proven particularly beneficial in geographically challenging regions, thereby revolutionizing communication paradigms [1], [2], [3], [4].

However, UAV networks (UAVNs), which frequently operate on an ad-hoc basis, can be vulnerable to various wireless attacks, including jamming and eavesdropping. This vulnerability is further exacerbated by UAVNs' critical role in security, emergency, and warfare communications, making them attractive targets for numerous threats. The open nature of these networks intensifies their susceptibility to jamming attacks, which can lead to a significant degradation in communication quality or even to a complete denial of service (DoS) [5]. Such attacks raise serious concerns about scalability and integration of UAVs into broader communication networks, posing significant challenges to their operation and reliability [6], [7], [8].

To address these challenges, various anti-jamming solutions for UAVNs have been proposed. Among them, game-theoretic approaches were used to model the strategic interactions between UAVs and jammers. For instance, Cheng et al. [9] proposed a Bimatrix Stackelberg game approach for adaptive frequency selection. Similarly, Xu et al. [10] used a Bayesian-Stackelberg game for strategy optimization under incomplete information scenarios. Moreover, Han et al. [11] proposed a distributed UAV deployment strategy using game theory. Furthermore, Feng et al. [12] extended game-theoretic principles to multi-agent reinforcement learning scenarios to address the challenges of UAV-assisted communication in the presence of jamming. Additionally, Zhang et al. [13] explored collaborative multi-agent jamming deceiving techniques, where

UAVs work together to deceive jammers and improve communication performance.

Previous studies also leveraged artificial intelligence to enhance anti-jamming capabilities. For instance, Ma et al. [14] used reinforcement learning for dynamic power control in UAVNs, while Wu et al. [15] applied Q-learning for joint optimization in the frequency-motion-antenna domain. Furthermore, Gao et al. [16] employed deep Q-networks (DQNs) to develop anti-intelligent jamming strategies.

Cooperative strategies proved to be a promising solution in mitigating jamming threats. In one of the relevant studies, Yu et al. [17] proposed a two-step anti-jamming strategy that encourages cooperation among UAVs. Furthermore, in [18], Almasoud optimized UAV trajectories using a genetic algorithm to minimize jamming effects during data collection. Similarly, Su et al. [19] developed a cooperative method using a local altruistic game model and a distributed algorithm to optimize channel selection among UAVs, thereby significantly enhancing anti-jamming performance. The utilization of machine learning techniques for jamming detection and classification has also gained traction, with Li et al. [20] proposing a feature-and spectrogram-tailored machine learning approach for OFDM-based UAVs. Furthermore, Luo and Liu [21] investigated the use of intelligent approach jamming in UAV-assisted wireless communication systems. The integration of machine learning with software-defined networking for enhanced network metric prediction and cyberattack detection in UAV relay networks has also been explored by Agnew et al. in [22] and [23], respectively.

However, while these solutions address various aspects of UAVN jamming threats, significant challenges remain. Specifically, most existing approaches do not take into account heavy jamming scenarios, where a significant portion of the available spectrum channels is under attack, e.g., up to 70% of the spectrum, or the growing threat of sophisticated machine learning-driven jamming attacks, where the disrupting entity employs advanced algorithms, notably deep reinforcement learning, to learn the behaviour of its targets and maximize its jamming effectiveness.

Seeking to address these challenges, recent research has explored the potential of predictive models and advanced machine learning architectures for anti-jamming. For instance, Pourranjbar et al. [24] demonstrated the use of Recurrent Neural Networks (RNNs) to predict jammers' occupied channels, enabling proactive mitigation strategies against multiple jamming policies. In another relevant study, the same group further investigated the use of RNNs for proactive channel prediction as a defence and offence strategy in tactical wireless networks [25]. Furthermore, using the power of a Transformer encoder within a deep reinforcement learning (DRL) framework, Xu et al. [26] investigated how Transformers could improve state space representation for anti-jamming optimization. The need for robust anti-jamming in the face of evolving threats has also driven investigations into distributed multi-agent reinforcement learning for cooperative learning [27], [28], particularly in scenarios where secure communication links might not be guaranteed [29]. Additionally, the combination of deep learning with distributed techniques has also shown promise, with the results showing improved environmental awareness and adaptive behaviour [30], [31]. However, these innovative methods have not been specifically tailored to the challenging environments faced by UAVNs, particularly in the presence of heavy jamming and sophisticated machine learning-based jamming attacks. The existing literature often overlooks the complexities arising from high jamming densities, where a significant portion of the available spectrum is under attack. Moreover, the dynamic and adaptive nature of machine learning-driven jamming attacks necessitates the development of proactive and intelligent anti-jamming solutions that can anticipate and counteract evolving threats.

Building upon these advancements, in the present study, we propose a novel anti-jamming approach that employs the superior predictive capabilities of Transformer models. Specifically, we introduce a Transformer-based module designed to predict the behaviour of multiple jammers, including those employing DRL-driven strategies. This method significantly differs from existing approaches by emphasizing real-time adaptability and proactive threat mitigation in high-density jamming environments. This predictive capability enables proactive countermeasures, which is a crucial step in securing UAVNs against emerging threats.

Our Transformer module is designed to operate both offline and online, providing the flexibility needed for dynamic UAV environments. While offline training on historical jamming patterns prepares the module, online adaptation allows real-time adjustment. We integrate these Transformer-based predictions with a pseudo-random (PR) channel selection algorithm in a robust hybrid approach. This combination benefits from the flexibility and unpredictability of randomization and the adaptability and intelligence provided by our predictive model. As a result, our offline and online Transformer-based method demonstrates significant performance advantages, with the online model achieving success rates of up to 70% and maintaining a high prediction accuracy of 75%, even within a complex environment characterized by the presence of a heavy jamming scenario and the use of diverse jamming attack policies by the disrupting entities.

Finally, to validate the effectiveness of our approach, we conduct rigorous simulations, with the results demonstrating its superiority compared to existing benchmarks. Specifically, the results of our simulations showcase significantly improved transmission success rates and prediction accuracy, highlighting how our method is better equipped to handle sophisticated threats and secure UAVNs in real-world operational environments.

The remainder of the article is organized as follows. In Section I, we present the modeling of the system, followed by

a detailed explanation of our proposed method in Section II. Section III discusses the outcomes of our approach in the results and analysis section. Finally, in Section IV, we draw conclusions, summarizing our findings and suggesting directions for further research.

II. SYSTEM MODEL

In this study, we propose a sophisticated communication system involving legitimate UAVs and a combination of terrestrial and non-terrestrial networks under heavy jamming attacks. The system comprises three legitimate UAVs, which are mobile and communicate with a terrestrial network composed of three nodes and a non-terrestrial network with a HAPS. All legitimate UAVs are geographically spread to prevent physical collision; yet wireless communication interference is possible. The spectrum of the communication links between legitimate UAVs and the nodes, or HAPS, is divided into N orthogonal channels. Furthermore, the communication process is divided into equal timeslots. There is no direct communication between UAVs, and they do not operate on a scheduling basis. Instead, to select unique channels, each UAV utilizes a pseudo-random (PR) channel selection pattern that is generated using a pseudo-random number generator, resulting in a deterministic yet seemingly random selection of channels unique to each UAV. PR patterns are established before deployment, ensuring that no two UAVs will attempt to transmit on the same channel during a given timeslot. This approach eliminates the possibility of interference, which occurs when multiple UAVs transmit on the same channel simultaneously. This approach rules out interference, which occurs when two or more UAVs choose the same channel for data transmission.

All channels between all elements of the system are reciprocal and follow a Rayleigh fading model, accounting for random fluctuations in signal amplitude and phase experienced in air-to-ground and air-to-HAPS links.

As shown in Fig. 1, our system includes multiple jammers with distinct behaviors. A DRL-powered smart jammer uses a DQN and interacts with the environment to learn an efficient attack policy. Additionally, a sweeping jammer targets one frequency channel at a time, while a comb jammer targets a group of three different frequency channels for T_{comb} timeslots, after which it attacks a different set of frequency channels and so on.

We assume that all elements in our system can sense the spectrum while continuing their transmission or attacking tasks. This enables them to have information about the spectrum activity of the last part of the timeslot before the start of the next one. A legitimate UAV's transmission is successful when its interactions with the node and the HAPS are not jammed.

In Section III, we present our proposed method to address the challenges identified in this system model.

III. PROBLEM FORMULATION

This section delineates the problem addressed in this study–namely, the issue of multiple jammers launching various types of attacks. In the environment outlined in Section II, each legitimate UAV follows a predefined PR channel selection agreement to counteract the smart jammer's attacks when communicating with the HAPS. In addition, the UAVs aim to thwart the sweeping and comb jammer attacks targeting their link with terrestrial nodes.

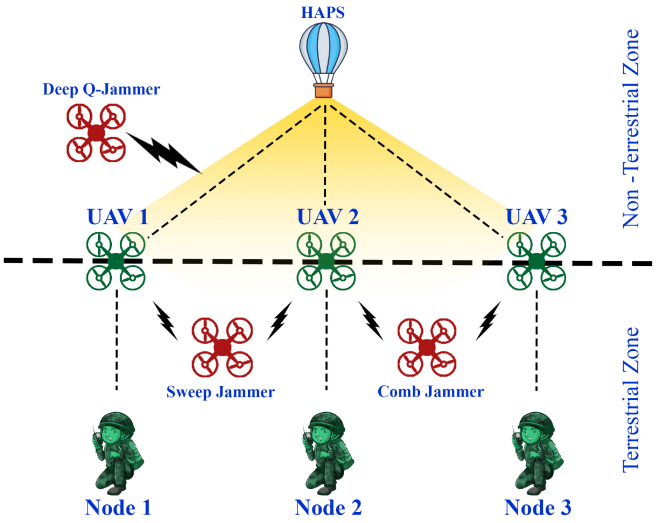

FIGURE 1. Diagram of the proposed communication system involving UAVs, nodes, jammers, and HAPS.

However, the smart jammer employs Deep Q learning to take action and uses the Markov Decision Process (MDP) to get the optimal behaviour. This MDP is characterized by the tuple $(\mathbb{S}, \mathbb{A}, P_a, R_a)$ outlined below.

- $\mathbb{S}$ represents the state space. At time-slot t, $s[t] = [c_1[t], c_2[t], .., c_N[t]] \in \mathbb{S}$ denotes the environment's state, where $c_i[t]$ is 0 if channel i is vacant or 1 otherwise.
- $\mathbb{A} = \mathbb{A}' \times \mathbb{A}' \times \mathbb{A}'$, where $\mathbb{A}' = \{1, 2, ..C\}$, C being the number of spectrum channels, constitutes the action space for the smart jammer. It simultaneously attacks three channels at time slot t. This action is defined as $a_j[t] = [c_j^1[t], c_j^2[t], c_j^3[t]]$, where $c_j^k[t] \in \mathbb{A}, k \in \{1, 2, 3\}$ is the k^{th} channel attacked by the jammer at time slot t.
- $P_{a_j}(s[t], s[t+1])$ is the transition probability between state $s[t]$ and $s[t+1]$ after the smart jammer takes action a_j.
- $R_j(s[t], a_j[t])$ is the reward the smart jammer receives for taking action a_j in state $s[t]$. At time slot t, the jammer receives a reward of 1 for each successfully jammed channel. Otherwise, it receives no reward. The reward for this jammer is given by Eq. (1):

$$R_i\big(s[t], a_j[t]\big) = \sum_{i=0}^{N_u} \chi_{j,i} \tag{1}$$

where

$$\chi_{j,j} = \begin{cases} 1 \text{ if } a_i[t] \in a_j[t], \\ 0 \text{ otherwise.} \end{cases} \tag{2}$$

The environment also contains two additional jammers. A sweeping jammer selects channel $c_{sj}[t] \equiv c_{sj}[t-1] \,(\mathrm{mod}\, N)$, and a comb jammer simultaneously attacks three channels as follows: $c_{cj}[t] = [c_{cj}^1[t], c_{cj}^2[t], c_{cj}^3[t]]$, maintaining its choice for T_{comb} time slots. Then, it targets a different random set of frequency channels and so on.

IV. PROPOSED METHOD

In this study, we introduce a novel method that uses a PR algorithm for initial channel selection, augmented with a Transformer module to predict the jammed channels in the subsequent timeslot. To combat these diverse jammers, we propose a novel anti-jamming approach built around two key components.

First, UAVs employ a PR channel selection algorithm for initial defence. Second, we augment this initial strategy with a Transformer Encoder-Decoder architecture to predict the jammed channels in the subsequent timeslot. The encoder portion is adept at capturing intricate relationships within sequential data, making it ideal for the analysis of historical jamming patterns. The decoder then uses the encoded information to generate a prediction of the jamming sequence for the next timeslot.

Transformers are a powerful deep-learning architecture known for their ability to model sequential relationships in data. This ability of Transformers is underpinned by a core mechanism called *'attention'*, which allows the model to focus on the most relevant parts of the input sequence when making predictions. In our context, the Transformer *'attends'* to specific moments in the jamming history to predict future jammer behaviour. Specifically, the Transformer's attention mechanism excels in finding patterns within historical data. In our system, this allows the model to identify correlations between the UAVs' PR choices and the jammers' subsequent attacks, thereby facilitating a proactive defence strategy.

This model is trained on data entries generated from an environment where only the PR algorithm is used. This training enables the model to predict the jammed channels relative to the choices made by legitimate UAVs using the PR algorithm. Therefore, when we employ the prediction, we are essentially predicting how the smart jammer reacts to the PR algorithm. We also predict the channels of the sweep and comb jammers.

The Transformer model operates on a sequence of past jamming states as input. Each input is represented as a binary vector of length C, where 1 indicates a jammed channel and 0 represents a free channel. The output of the model is also a binary vector of length C, with each element predicting the presence (1) or absence (0) of a jammer on the corresponding channel in the next timeslot.

Overall, our Transformer model features a customized embedding layer designed to process binary sequences representing jamming states. This differs significantly from traditional Transformer models, which typically operate on sequences of words or tokens. This adaptation enables our model to effectively capture the dependencies and patterns inherent in binary jamming data.

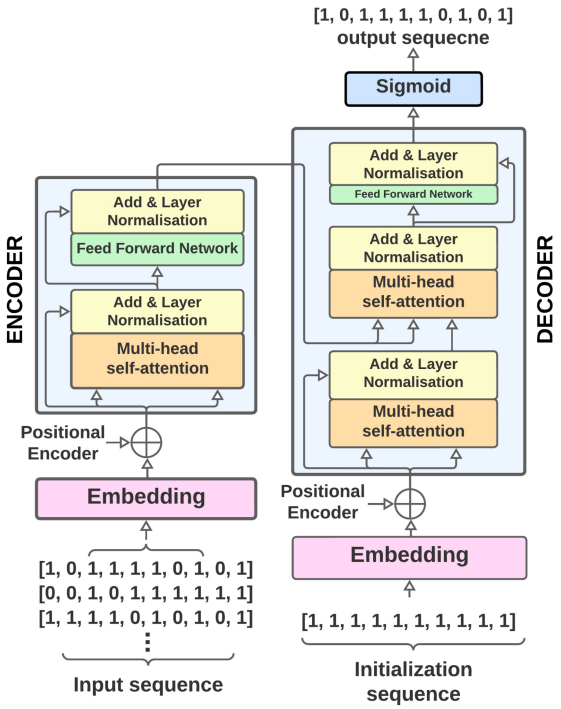

FIGURE 2. Encoder-Decoder Transformer module architecture for the jamming position prediction.

A. TRANSFORMER WORKFLOW

As shown in Fig. 2, the Transformer processes the jamming history through a series of steps to predict future jammer behaviour:

- *Input Embedding:* The input to the Transformer is a sequence of past jamming states, representing each state as a binary vector. The embedding layer transforms these binary vectors into a rich vector representation, capturing the semantic meaning of each jamming state.
- *Positional Encoding:* Since the Transformer architecture does not inherently capture the order of the input sequence, positional encoding is added to the embedded representation. This encoding injects information about the relative or absolute position of each jamming state in the sequence, allowing the model to understand the temporal dynamics of the jamming patterns.
- *Encoder Self-Attention:* The core of the Transformer is the self-attention mechanism. In the encoder, the self-attention layer allows the model to weigh the importance of different positions in the input sequence when processing a particular position. This enables the model to capture complex dependencies and relationships between different jamming states in the history.
- *Encoder Feedforward Network:* After the self-attention layer, a feedforward network further processes the encoded representation. This network consists of two linear transformations with a ReLU activation function

in between. It helps to capture more complex patterns in the data.

- *Decoder Initialization and Embedding:* The decoder starts with an initial sequence (e.g., a 'start' token) and embeds it into a continuous representation, similar to the encoder.
- *Decoder Self-Attention and Encoder-Decoder Attention:* The decoder also has a self-attention layer to focus on the relationships within the generated output sequence. Additionally, it has an encoder-decoder multi-head attention layer that allows it to attend to relevant parts of the encoded input sequence, helping it generate accurate predictions.
- *Decoder Feedforward Network and Output:* Similar to the encoder, the decoder has a feedforward network that performs the final processing step where normalization again ensures stability, and the sigmoid function maps the output values between 0 and 1 for probability-like predictions.

Our method encompasses two distinct variants to balance robustness and adaptability: offline and online. The offline variant is trained on a pre-existing dataset of jamming patterns and does not update its parameters during deployment. It provides robust initial performance based on the patterns it learned during training. In contrast, the online variant continuously adapts to the changing jamming environment. It maintains a memory of recent jamming data and periodically retrains the Transformer model, refining its predictions based on real-time observations.

To illustrate, consider a scenario where a new jamming strategy emerges after the initial deployment of the offline model. The offline model would be unable to adapt to this new strategy, potentially leading to a decrease in performance. However, the online model would detect the shift in jamming behaviour, retrain on the updated data, and adjust its predictions accordingly, maintaining a high level of anti-jamming effectiveness.

B. OFFLINE METHOD

The offline variant of our method begins with the pre-training of a Transformer model. This training is conducted on simulated data where UAVs employ the PR channel selection algorithm in the presence of jammers. Once the anti-jamming scenario begins, the system maintains a memory window of $T_{features}$ timeslots. Initially, this window is filled with placeholder values (e.g., ones) to provide a neutral baseline for the Transformer predictions. The Transformer module takes this memory window as input to predict the jammer positions in the next timeslot.

C. ONLINE METHOD

The online variant of our method builds upon the offline variant's foundation. It also starts with a pre-trained Transformer model. However, this model is continuously refined during live operation. The system maintains a training memory of length T_{tuning} storing recent jamming data to achieve this. Once T_{tuning} is reached, the Transformer model is retrained on these accumulated data, thereby allowing the model to adapt to the jammers' changing behaviors.

Algorithm 1 Offline Version of the Proposed Algorithm

```
Initialize UAVs
Initialize the Transformer model
Simulate UAVs behaviors using the PR channel selection algorithm under jammer attacks
Construct a training dataset with jammer channel positions from the simulations
Train the Transformer model with the training dataset
while UAVs are operational do
    The Transformer model takes the current memory window of jammed channels with length T_features as input // Initially, the memory window is filled with placeholder values
    The Transformer model predicts the next jammed channel information
    UAVs select their actions from the predicted free channels by the Transformer model
    Update the memory window with the last jamming information // Maintain a memory window of the past T_features timeslots for the prediction operation
end while
```

Algorithm 2 Online Version of the Proposed Algorithm

```
Initialize UAVs
Initialize the Transformer model
Initialize the training dataset
while UAVs are operational do
    The Transformer model takes the current memory window of jammed channels with length T_features as input // Initially, the memory window is filled with placeholder values
    if T_tuning is reached then
        Train the Transformer model on the training dataset of length T_tuning
    end if
    The Transformer model predicts the next jammed channel
    UAVs select their actions from the predicted free channels by the Transformer model
    Update the training dataset with ongoing jammed channels // Maintain a training memory of length T_tuning for the tuning operation
    Update the memory window with the last jamming information // Maintain a memory window of the past T_features timeslots for the prediction operation
end while
```

D. DECISION MAKING

Both offline and online variants of our method employ the same core decision-making logic. Before applying the

sigmoid function, the UAVs analyze the Transformer model's output and prioritize channels with the lowest predicted values. This strategy enhances robustness, as these channels are most likely to be free of jamming activity. Therefore, by focusing on the channels the Transformer predicts to be clear, the UAVs proactively mitigate the impact of unexpected jamming events.

In Section V, we compare offline and online versions of our algorithm in terms of success rate and predictive precision.

E. COMPUTATIONAL COMPLEXITY

The computational complexity of our Transformer-based method during the testing (prediction) phase is influenced by the following factors:

- Input Dimensionality: The Transformer receives an input with dimensions $T_{features} \times N_{channels}$. Transformers generally exhibit quadratic complexity with respect to the total number of features $[O((T_{features} \times N_{channels})^2)]$.
- Transformer Attention Heads (H): The number of attention heads in the Transformer model directly impacts complexity. We can approximate the relationship as linear $[O(H)]$.

Combining the dominant factors, we estimate complexity as approximately $O((T_{features} \times N_{channels})^2 \times H)$. In the scenario where $T_{features}$ and H are considered constant, the complexity scales quadratically with the number of channels $[O(N^2_{channels})]$.

Crucially, offline training can be performed with powerful computational tools, making its complexity less of a concern for real-time deployment. Our method's specific parameter configuration ($T_{features} = 13, N_{channels} = 10, H = 2$) leads to a low computational footprint, making it suitable for onboard processing on UAVs.

While the computational complexity of the Transformer model is a consideration, it is well within the capabilities of modern UAV hardware. The feasibility of deploying Transformer models on UAVs for real-time tasks has been demonstrated in recent research [32]. Recent advancements in embedded systems and edge computing have led to the development of powerful and energy-efficient processors that can handle the computational demands of deep-learning models like Transformers. Moreover, techniques such as model quantization [33] and pruning [34] can be employed to further reduce the model's size and computational requirements without significantly sacrificing performance.

V. SIMULATION RESULTS AND ANALYSIS

In this section, we present the results of our simulations and compare the different methodologies employed in our research. Our aim is to evaluate the effectiveness of our proposed Transformer-based anti-jamming techniques as compared to a baseline PR channel selection approach and other machine learning paradigms. We begin by describing the experimental setup used to evaluate our methods presented in Section IV.

TABLE 1. Simulation parameters.

Description	Value
Number of channels	[10, 12, 14]
Number of agents	3
Number of jammers	3
Jamming types	'smart', 'comb', 'sweeping'
Comb jammer depth : T_{comb}	50 timeslots
Transformer embedding size	64
Number of training epochs	100 epochs
Number of episode timeslots	11500 timeslots
Tuning period : T_{tuning}	2048 timeslots
Prediction window : $T_{features}$	13 timeslots

A. EXPERIMENTAL SETUP

Our experimental setup involves a complex scenario with 10 channels, 3 legitimate users, and 3 jammers: a smart jammer employing a DQN, capable of attacking 3 channels simultaneously, a comb jammer attacking 3 channels every T_{comb} timeslots, and a sweeping jammer attacking 1 channel per timeslot. This diversity of jammers allows us to evaluate the robustness of our method against various and simultaneous jamming threats. Table I provides a summary of the considered simulation parameters. Additionally, we investigate the impact of varying the number of channels, testing our methods' performances on environments with 10, 12 and 14 channels. This choice reflects a compromise between evaluating the method's effectiveness under different jamming densities and maintaining a realistic number of channels for UAV communications.

The simulations are conducted on a PC with the following specifications: 13th Gen Intel Core i7-13700 CPU with 16 cores at 2.10 GHz, NVIDIA RTX A2000 GPU with 12 GB GDDR6 memory, and 32 GB DDR5 RAM. The software environment included CUDA Version 11.8, Python 3.11.5, and PyTorch 2.1+cu118.

B. METHODOLOGIES

Our simulations employ the offline (TR-OFF) and online (TR-ON) variants of the method introduced in Section IV. Both rely on a Transformer model, with the key difference being the online variant's real-time tuning process. To generate training data, we initially use a PR channel selection algorithm. To provide a comprehensive evaluation, we compare the two variants of our method against the following three benchmarks:

- *DQ:* Legitimate users independently use their DQNs to learn channel selection strategies that mitigate jamming, with rewards based on their success in avoiding the attacks.
- *Offline LSTM (LSTM-OFF):* An LSTM module predicts jammer positions and users select channels accordingly. This serves as a comparison to evaluate the Transformer's effectiveness in a similar time-series prediction setting.

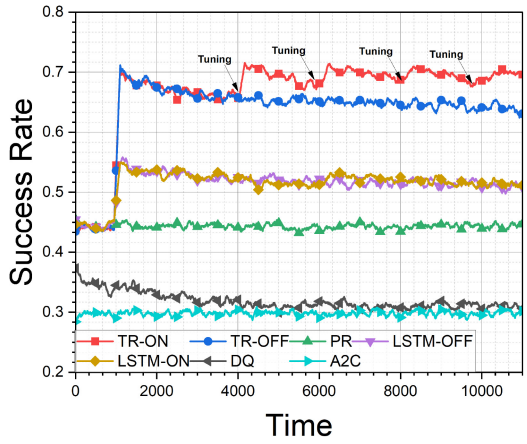

FIGURE 3. Success rate for the different algorithms.

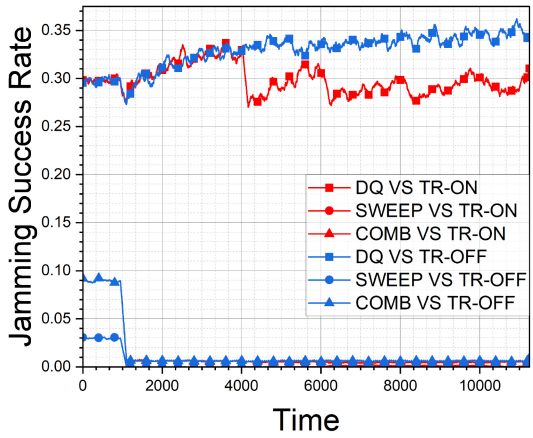

FIGURE 4. Jammers success rate against both online and offline Transformer methods.

- *Online LSTM (LSTM-ON):* Similar to LSTM-OFF, yet with the model periodically fine-tuned to parallel the TR-ON approach.

C. COMPARATIVE ANALYSIS

The results of our simulations providfe valuable insights into each method's performance. In this section, we present the results of a comprehensive comparative analysis, examining success rates, jammer performance against our Transformer-based methods, prediction accuracy, and the impact of varying channel count. Our goal is to showcase the efficiency of our proposed methods for predicting and mitigating heavy jamming attacks.

1) SUCCESS RATE COMPARISON

We begin by comparing the success rates of TR-ON, TR-OFF, and other benchmark methods. The success rate is the number of timeslots where legitimate UAVs do not get jammed, divided by the total number of timeslots.

Fig. 3 illustrates the success rates over time for seven distinct methods: TR-ON, TR-OFF, A2C, PR, LSTM-OFF, LSTM-ON, and DQ. The TR-ON method consistently outperforms the other methods, achieving a success rate of 70% by the end of the observed period. This demonstrates its high effectiveness in mitigating jamming attacks within a heavily jammed and dynamic environment. The TR-OFF method also exhibits commendable performance, reaching a success rate over 60%. However, its lack of a tuning mechanism makes it 10% less effective than the TR-ON method. Each tuning period, denoted as T_{tuning}, boosts TR-ON's success rate, indicating its adaptability to the changing behaviour of the DQ-smart jammer. The A2C method, on the other hand, maintains a consistent yet comparatively lower success rate of approximately 30%, highlighting the limitations of this reinforcement learning approach in this scenario. Furthermore, owing to the random nature of the channel selection algorithm, the PR method shows a steady success rate of ca. 45%. Next, LSTM-OFF and LSTM-ON methods demonstrate relatively lower performance than the Transformer-backed algorithms, with success rates slightly above 50%. Finally, the DQ-based algorithm exhibits the poorest performance, with success rates falling below 40% and approaching 30% by the end of the simulation period. This is due to the DQ-jammer's ability to learn and adapt to the behaviour of the DQ-powered legitimate users.

Fig. 3 clearly highlights the superior performance of the Transformer-backed methods, particularly TR-ON, in terms of success rate over time.

2) JAMMER PERFORMANCE

This section analyzes the performance of the DQ, comb, and sweep jammers when faced with the TR-OFF and TR-ON anti-jamming algorithms. Fig. 4 depicts the success rates of these jammers against three legitimate users.

The comb and sweep jammers are completely thwarted by the Transformer-based models, with their success rates plummeting to zero. This finding demonstrates that both TR-ON and TR-OFF can effectively learn and counter the patterns of these jammers from the training data.

By contrast, the smart jammer exhibits a more complex dynamic. Its success rate against the offline model (DQ vs. TR-OFF) steadily increases over time, reaching a peak of 35%. However, against the online model (DQ vs. online), its success rate fluctuates below 30%. This cyclical pattern reflects the TR-ON model's tuning mechanism, which continually forces the smart jammer to adapt. Each tuning period temporarily disrupts the smart jammer. The latter then subsequently improves until the next tuning cycle.

These observations highlight the TR-ON model's adaptability to changing jamming strategies. Despite a temporary decrease in prediction accuracy during the tuning period, the TR-ON model's ability to learn and adapt to new patterns used by the smart jammer ultimately results in its lower overall jamming success rate.

3) ACCURACY OF TRANSFORMER MODELS VS. LSTM

This section presents the results of a comparative analysis of the prediction accuracy of TR-ON, TR-OFF, and the

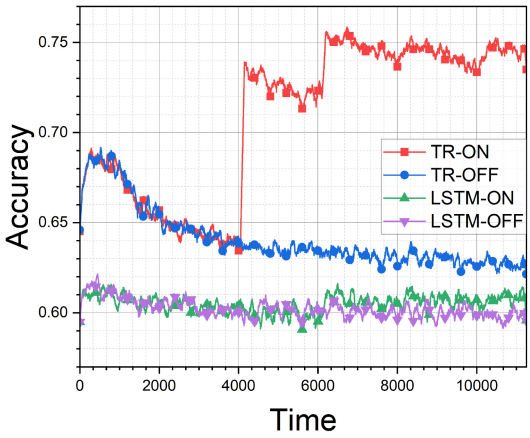

FIGURE 5. Prediction accuracy comparison of the Transformer vs. LSTM models.

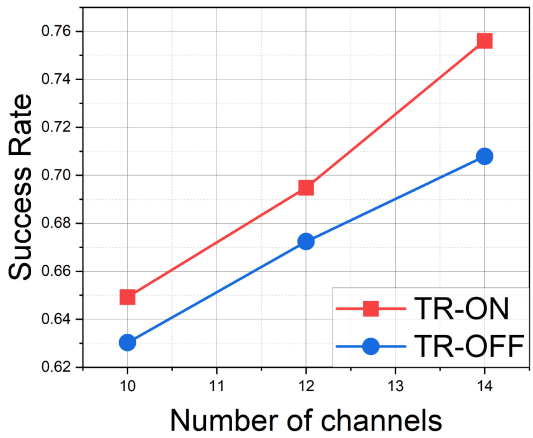

FIGURE 6. Success rate as a function of number of channels.

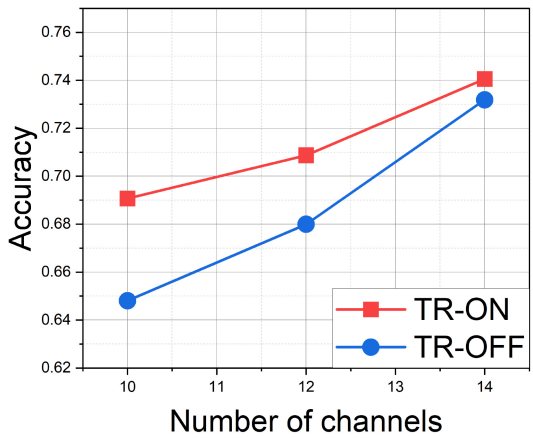

FIGURE 7. Prediction accuracy as a function of number of channels.

LSTM-backed models. This comparison is instrumental in understanding each method's relative strengths and weaknesses in predicting the jammers' positions.

Fig. 5 reveals distinct performance characteristics of the Transformer vs. LSTM models. The TR-ON and TR-OFF models consistently demonstrate a superior prediction accuracy as compared to their LSTM counterparts.

The TR-ON model exhibits the highest accuracy overall, which significantly increases around the 4000 mark on the time axis and remains high thereafter. By contrast, the LSTM-OFF model has the lowest accuracy throughout the observed period. While the LSTM-ON model shows improvement towards the end, approaching the TR-OFF model's accuracy, it still falls short of the superior performance of the TR-ON model.

4) IMPACT OF THE NUMBER OF CHANNELS

We examine the impact of increasing the number of channels on the performance of our Transformer-based anti-jamming methods. In this test, we deliberately maintain a high jamming density, ensuring at least half the channels are under potential jamming threat. We specifically choose three channel counts: 10, 12, and 14. This equates to jamming ratios of 0.70, 0.58, and 0.5, respectively, keeping the number of legitimate UAVs and jammers constant. Fig. 6 illustrates how the average success rate of legitimate UAVs changes with an increase in the number of channels. We observe a general trend of increasing success rates for both the TR-ON and TR-OFF models as the number of channels increases. This indicates that, in the event of jamming threats, a larger pool of available channels benefits UAV communications. Importantly, the TR-ON model consistently outperforms the TR-OFF across all channel counts, emphasizing the value of its real-time adaptability. This advantage becomes even more pronounced as the number of channels scales. For example, at 10 channels, the difference is minimal, with TR-ON achieving 65% and TR-OFF 63%; however, at 14 channels, TR-ON achieves a success rate of ca. 76%, while TR-OFF remains around 70%. This highlights that, in complex environments with more channels, the ability to adapt to changing jamming strategies becomes crucial. These results convincingly demonstrate the effectiveness of our Transformer-based approaches, particularly the online-adaptive TR-ON model, in securing UAV communications against adversarial jamming.

In Fig. 7, we observe a positive correlation between the number of channels and the prediction accuracy of our Transformer-based models. The increased dimensionality of the data allows both the TR-ON and TR-OFF models to better learn the underlying patterns of jammer behaviour. Interestingly, with an increase of the number of channels to 14, with a corresponding jamming ratio of 50%, the difference in accuracy between the TR-ON and TR-OFF models diminishes and amounts to 1%. This suggests that, when the environment provides a sufficient number of free channels, the pre-trained knowledge of the TR-OFF model becomes sufficient to achieve prediction accuracy levels comparable to those afforded by the online-adaptive TR-ON method.

5) TRANSFORMER MODEL TRAINING AND CONVERGENCE

The Transformer model is trained using the AdamW optimizer with a learning rate scheduler. The training process

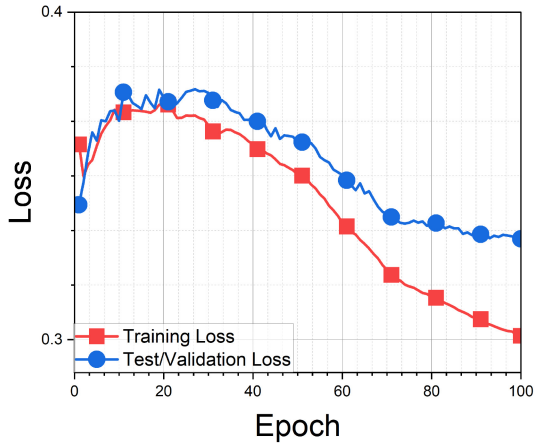

FIGURE 8. Transformer Model Convergence.

aims to minimize the binary cross-entropy loss between the predicted and actual jamming states. The convergence behaviour of the model is illustrated in Fig. 8. Both the training and validation losses initially increase. In the initial phase of training (epoch < 10), the observed increase in loss can be attributed to the random initialization of the model's parameters and the optimizer's initial exploration of the parameter space. Furthermore, since the data is imported batch by batch, the model has not yet seen the entire dataset. As the model begins to capture the underlying patterns in the data and visits more of it across epochs, the optimizer's updates become more refined, leading to the subsequent decrease in both training and validation loss. The training loss continues to decrease steadily over the epochs, ultimately reaching a value of approximately 0.3 at 100 epochs. The validation loss, while also decreasing, experiences more fluctuations and plateaus around 0.35, slightly above the final training loss. The alignment of the two curves in later epochs suggests the model is generalizing well and not overfitting. The slight remaining gap between training and validation loss indicates the potential for further marginal improvement in generalization performance. Overall, the consistent downward trend and proximity of the curves demonstrate the model's effectiveness in learning the underlying patterns of the data and its ability to generalize to unseen jamming scenarios.

6) ANALYSIS AND TRADE-OFF

Our comparative analysis underscores the effectiveness of Transformer-based anti-jamming methods and reveals key trade-offs between the TR-ON and TR-OFF models.

The TR-ON model, with its real-time adaptability, consistently outperforms all other methods in terms of success rate, demonstrating resilience in dynamic, jammer-rich environments. This superior performance is due to its ability to continuously learn and adapt to the evolving tactics of intelligent jammers. Unlike the TR-OFF, which relies on pre-trained knowledge and struggles to adjust to novel or unexpected jamming strategies, the TR-ON actively updates its model parameters by retraining on recent jamming data. This continuous learning allows the TR-ON to identify and respond to shifts in jammer behaviour, which is crucial when facing intelligent jammers employing deep reinforcement learning. As a result, the TR-ON maintains a higher success rate, staying ahead of the jammer's evolving tactics. Interestingly, the distinction in success rates grows with an increase in the complexity of the environment (i.e., a higher number of channels).

Of note, both Transformer models successfully thwart the comb and sweep jammers, highlighting their effectiveness in countering predictable jamming patterns. Furthermore, they exhibit a consistently higher prediction accuracy as compared to the LSTM-based benchmarks, underscoring the Transformer architecture's superiority for time-series analysis in jamming scenarios.

An important consideration is the effect of channel count. By providing more opportunities for legitimate communication to evade jamming, an increase in the number of channels generally boosts the success rate of both Transformer models. Interestingly, the difference in their prediction accuracy diminishes at higher channel counts, especially when the jamming ratio drops to 50%. This suggests a potential point where the TR-OFF model's pre-trained knowledge becomes sufficient under less congested conditions.

This observation highlights a critical trade-off: computational complexity. While the TR-ON delivers superior performance, especially in dynamic and heavily jammed environments, it requires a robust computational setup. This becomes particularly important for real-world UAVs with onboard processing units for training tasks. However, despite being computationally less demanding, the TR-OFF offers a slightly lower success rate overall.

7) ROBUSTNESS AGAINST CHANNEL IMPERFECTIONS AND NOISE

In this part of our performance analysis, we fix the number of channels to $N_c = 10$. Our system demonstrates adaptability to varying levels of channel imperfections and noise. We use a threshold-based strategy for channel detection, ensuring accurate identification of jammed channels. This approach considers channels with received base-band signal amplitude exceeding a threshold (Θ) jammed. However, the choice of this threshold presents a trade-off between the probability of false alarm (P_{fa}), where noise is mistaken for jamming, and the probability of miss detection (P_{md}), where a jamming signal goes undetected. Following [24], Eq. (3) and Eq. (4) show that P_{fa} depends solely on the threshold value (Θ) and the noise power (σ^2):

$$P_{\text{fa}} = Q\left(\frac{\Theta}{\sigma}\right) \tag{3}$$

By contrast, P_{md} is also influenced by the jamming-to-noise ratio (JNR), which incorporates the jamming signal

TABLE 2. P_{md} for Different P_{fa} and *JNR* Values.

JNR/P_{fa}	**0.01**	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**
5	0.7082	0.4469	0.3097	0.2291	0.1745	0.1348
8	0.4264	0.1930	0.1093	0.07005	0.0474	0.0331
12	0.0490	0.0097	0.0035	0.0016	0.0008	0.0005
15	0.0005	$3.5e^{-5}$	$7.1e^{-6}$	$2.3e^{-6}$	$8.7e^{-7}$	$3.7e^{-7}$
18	$9.7e^{-9}$	$1.5e^{-10}$	$1.4e^{-11}$	$2.5e^{-12}$	$6.2e^{-13}$	$1.8e^{-13}$

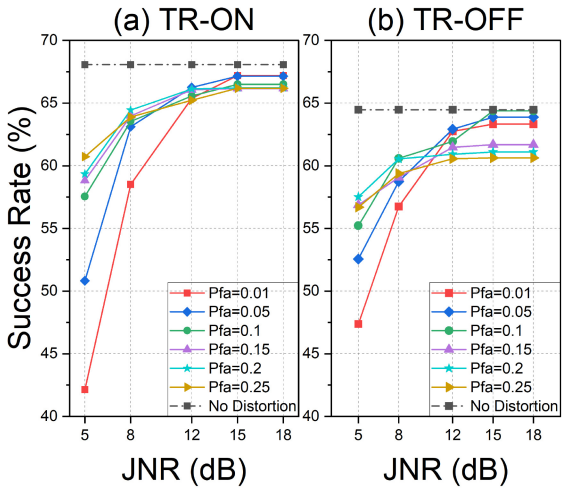

FIGURE 9. Effect of P_{fa} and P_{md} on the success rate.

power (η):

$$P_{\text{md}} = Q\left(\frac{\eta - \Theta}{\sigma}\right) \tag{4}$$

As illustrated in Table II, a higher *JNR* (stronger jamming signal) generally translates to a lower P_{md}, making the jamming signal easier to detect. Conversely, a lower *JNR* makes detection more challenging and can increase P_{md}.

Simulating realistic jamming environments JNR is a dynamic parameter reflecting the real-world challenges of UAV communication. While moving, UAVs experience variations in their distance from jammers. Higher *JNRs* occur when UAVs are closer to jammers due to the stronger jamming signal strength. Conversely, lower *JNRs* represent situations where UAVs are farther away and the jamming signal weakens. To evaluate how our system performs under varying levels of jamming intensity, we simulate these dynamic *JNR* conditions.

Fig. 9 illustrates the success rates of both the TR-ON (Fig. 9 (a)) and TR-OFF (Fig. 9 (b)) anti-jamming models under varying degrees of channel imperfections. The impact of *JNR*, P_{md}, and P_{fa} is clearly visible. The ideal scenario without imperfections (dotted line) sets the benchmark for the other scenarios, as it represents the maximum achievable success rate. In line with our expectation, the success rates decrease with increasing P_{md} and P_{fa}. For both models, high *JNR* levels (above 12 dB) lead to success rates closer to the ideal case, as lower P_{md} allows for a more reliable jamming detection.

Interestingly, the success rates remain relatively high even with increased P_{fa} at high *JNRs*. This suggests that both models are more robust to false alarms than miss detections. A missed detection has a greater detrimental effect, as it means a jammed channel remains unidentified. Importantly, the TR-ON model consistently outperforms the TR-OFF model across various combinations of *JNR*, P_{md}, and P_{fa}, demonstrating its superior ability to mitigate jamming within imperfect channel conditions.

VI. CONCLUSION

This study demonstrates the effectiveness of Transformer-based models for mitigating heavy jamming attacks within dense UAV and HAPS communication systems. Combining pseudo-random channel selection with a Transformer's adaptive prediction capabilities, our hybrid methodology offers distinct advantages over other approaches. We design the Transformer to effectively operate in both offline and online modes.

In dynamic environments with smart jammers, the online Transformer (TR-ON) surpasses other methods, including the offline Transformer (TR-OFF), LSTM models, DQN-based strategies, and basic PR techniques. The TR-ON demonstrates superior success rates and prediction accuracy, particularly with an increase in the number of channels. It also successfully thwarts comb and sweep jammers. Interestingly, with a large number of channels and lower jamming density, the accuracy gap between TR-ON and TR-OFF diminishes, highlighting a potential trade-off between computational efficiency and adaptability. Our analysis highlights the system's robustness to channel imperfections and noise. Of note, both models demonstrate a higher tolerance for false alarms, but missed detections have a more significant negative impact on success rates. The TR-ON maintains its superior performance across various noise scenarios.

Overall, our results provide robust evidence that Transformer models are powerful tools for enhancing UAV communication robustness against jamming threats. However, it is important to acknowledge some potential limitations to our approach. The performance of both TR-OFF and TR-ON is contingent on the quality and representativeness of the training data. If the training data does not adequately capture the diversity of potential jamming strategies, the models may struggle to generalize to unseen scenarios, resulting in reduced effectiveness. Furthermore, the Transformer model itself could be a target for adversarial attacks. Adversaries could attempt to manipulate the input data or exploit vulnerabilities in the model's architecture to mislead its predictions. For instance, a jammer might initially use a predictable pattern that the model learns, only to change its strategy later, causing the model to make incorrect predictions. This could potentially compromise the UAV's anti-jamming capabilities.

Future research should focus on cooperative UAV strategies, latency optimization, and integration with broader

spectrum-sharing protocols. These advancements would further strengthen the development of secure and adaptable UAV anti-jamming systems.

REFERENCES

[1] F. Hsieh, F. Jardel, E. Visotsky, F. Vook, A. Ghosh, and B. Picha, "UAV-based multi-cell HAPS communication: System design and performance evaluation," in *Proc. IEEE Global Commun. Conf.*, 2020, pp. 1–6.

[2] L. Song, B. Di, H. Zhang, and Z. Han, *Aerial Access Networks: Integration of UAVs, HAPs, and Satellites*. Cambridge, U.K.: Cambridge Univ., 2023.

[3] J. Pelton, "High altitude platform systems (HAPS) and unmanned aerial vehicles (UAV) as an alternative to small satellites," in *Handbook of Small Satellites: Technology, Design, Manufacture, Applications, Economics and Regulation*. Cham, Switzerland: Springer, 2020, pp. 385–399.

[4] A. Pourranjbar, M. Baniasadi, A. Abbasfar, and G. Kaddoum, "A novel distributed algorithm for phase synchronization in unmanned aerial vehicles," *IEEE Commun. Lett.*, vol. 24, no. 10, pp. 2260–2264, Oct. 2020.

[5] H. Sedjelmaci, S. M. Senouci, and N. Ansari, "A hierarchical detection and response system to enhance security against lethal cyber-attacks in UAV networks," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 48, no. 9, pp. 1594–1606, Sep. 2018.

[6] G. K. Pandey, D. S. Gurjar, H. H. Nguyen, and S. Yadav, "Security threats and mitigation techniques in UAV communications: A comprehensive survey," *IEEE Access*, vol. 10, pp. 112858–112897, 2022.

[7] Y. Zhi, Z. Fu, X. Sun, and J. Yu, "Security and privacy issues of UAV: A survey," *Mobile Netw. Appl.*, vol. 25, pp. 95–101, Feb. 2020.

[8] V. Behzadan, "Cyber-physical attacks on UAS networks-challenges and open research problems," 2017, *arXiv:1702.01251*.

[9] L. Cheng et al., "Adaptive spectrum anti-jamming in UAV-enabled air-to-ground networks: A bimatrix Stackelberg game approach," *Electronics*, vol. 12, no. 20, p. 4344, 2023.

[10] Y. Xu et al., "A one-leader multi-follower Bayesian-Stackelberg game for anti-jamming transmission in UAV communication networks," *IEEE Access*, vol. 6, pp. 21697–21709, 2018.

[11] C. Han, A. Liu, K. An, G. Zheng, and X. Tong, "Distributed UAV deployment in hostile environment: A game-theoretic approach," *IEEE Wireless Commun. Lett.*, vol. 11, no. 1, pp. 126–130, Jan. 2022.

[12] Z. Feng, M. Huang, D. Wu, E. Q. Wu, and C. Yuen, "Multi-agent reinforcement learning with policy clipping and average evaluation for UAV-assisted communication Markov game," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 12, pp. 14281–14293, Dec. 2023.

[13] C. Zhang, Z. Xiao, L. Zhang, G. Liu, W. Zhang, and X.-G. Xia, "Collaborative multi-agent jamming deceiving for UAV-assisted wireless communications," in *Proc. Int. Wireless Commun. Mobile Comput. (IWCMC)*, 2024, pp. 549–554.

[14] N. Ma et al., "Reinforcement learning-based dynamic anti-jamming power control in UAV networks: An effective jamming signal strength based approach," *IEEE Commun. Lett.*, vol. 26, no. 10, pp. 2355–2359, Oct. 2022.

[15] Q. Wu, H. Wang, X. Li, B. Zhang, and J. Peng, "Reinforcement learning-based anti-jamming in networked UAV radar systems," *Appl. Sci.*, vol. 9, no. 23, p. 5173, 2019.

[16] N. Gao, Z. Qin, X. Jing, Q. Ni, and S. Jin, "Anti-intelligent UAV jamming strategy via deep Q-networks," *IEEE Trans. Commun.*, vol. 68, no. 1, pp. 569–581, Jan. 2020.

[17] J. Yu, Y. Gong, J. Fang, R. Zhang, and J. An, "Let us work together: Cooperative beamforming for UAV anti-jamming in space–air–ground networks," *IEEE Internet Things J.*, vol. 9, no. 17, pp. 15607–15617, Sep. 2022.

[18] A. Almasoud, "Jamming-aware optimization for UAV trajectory design and internet of things devices clustering," *Complex Intell. Syst.*, vol. 9, pp. 1–20, Aug. 2023.

[19] Y. Su, N. Qi, Z. Huang, R. Yao, and L. Jia, "Cooperative anti-jamming for UAV networks: A local altruistic game approach," 2021, *arXiv:2109.09467*.

[20] Y. Li et al., "Jamming detection and classification in OFDM-based UAVs via feature-and spectrogram-tailored machine learning," *IEEE Access*, vol. 10, pp. 16859–16870, 2022.

[21] S. Luo and X. Liu, "UAV intelligent approach jamming wireless communication system," in *Proc. 3rd Int. Conf. Neural Netw., Inf. Commun. Eng. (NNICE)*, 2023, pp. 427–432.

[22] D. Agnew, A. Del Aguila, and J. McNair, "Enhanced network metric prediction for machine learning-based cyber security of a software-defined UAV relay network," *IEEE Access*, vol. 12, pp. 54202–54219, 2024.

[23] D. Agnew, A. del Aguila, and J. McNair, "Detection of cyberattacks in an software-defined UAV relay network," in *Proc. IEEE Mil. Commun. Conf. (MILCOM)*, 2023, pp. 504–509.

[24] A. Pourranjbar, G. Kaddoum, and W. Saad, "Recurrent-neural-network-based anti-jamming framework for defense against multiple jamming policies," *IEEE Internet Things J.*, vol. 10, no. 10, pp. 8799–8811, May 2023.

[25] A. Pourranjbar, I. Elleuch, S. Landry-pellerin, and G. Kaddoum, "Defense and offence strategies for tactical wireless networks using recurrent neural networks," *IEEE Trans. Veh. Technol.*, vol. 72, no. 6, pp. 8278–8283, Jun. 2023.

[26] J. Xu, H. Lou, W. Zhang, and G. Sang, "An intelligent anti-jamming scheme for cognitive radio based on deep reinforcement learning," *IEEE Access*, vol. 8, pp. 202563–202572, 2020.

[27] Z. Yin, Y. Lin, Y. Zhang, Y. Qian, F. Shu, and J. Li, "Collaborative multiagent reinforcement learning aided resource allocation for UAV anti-jamming communication," *IEEE Internet Things J.*, vol. 9, no. 23, pp. 23995–24008, Dec. 2022.

[28] Z. Feng et al., "Approximating Nash equilibrium for anti-UAV jamming Markov game using a novel event-triggered multi-agent reinforcement learning," *Neural Netw.*, vol. 161, pp. 330–342, Apr. 2023.

[29] I. Elleuch, A. Pourranjbar, and G. Kaddoum, "A novel distributed multi-agent reinforcement learning algorithm against jamming attacks," *IEEE Commun. Lett.*, vol. 25, no. 10, pp. 3204–3208, Oct. 2021.

[30] N. Rao et al., "Joint optimization of jamming link and power control in communication countermeasures: A multiagent deep reinforcement learning approach," *Wireless Commun. Mobile Comput.*, vol. 2022, no. 1, pp. 1–18, 2022.

[31] I. Elleuch, A. Pourranjbar, and G. Kaddoum, "Deep cross-check Q-learning for jamming mitigation in wireless networks," *IEEE Wireless Commun. Lett.*, vol. 13, no. 5, pp. 1448–1452, May 2024.

[32] X. Sun et al., "Siamese transformer network: Building an autonomous real-time target tracking system for UAV," *J. Syst. Archit.*, vol. 130, Sep. 2022, Art. no. 102675.

[33] R. Krishnamoorthi, "Quantizing deep convolutional networks for efficient inference: A whitepaper," 2018, *arXiv:1806.08342*.

[34] D. Blalock, J. J. Gonzalez Ortiz, J. Frankle, and J. Guttag, "What is the state of neural network pruning?" in *Proc. Mach. Learn. Syst.*, vol. 2, 2020, pp. 129–146.