Cognitive Radio Anti-Jamming Develop Environment

Vincent Prado
Electric and Computer Engineering
University of Puerto Rico at Mayaguez
Mayaguez, Puerto Rico
vincent.prado@upr.edu

Pauline Wu
Information & Computer Sciences
University of Hawaii at Manoa
Honolulu, Hawaii
peihanwu@hawaii.edu

***Abstract*—Due to the lack of required infrastructure, the HF band is popular for military communications over long distances. This makes this band a popular target when it comes to jamming attacks. Our objective with this work is to create an HF environment to develop and test jamming and anti-jamming technology in a competitive environment using Simulink. This environment provides the option to change the environmental conditions of the HF (High Frequency) channel and will let the user choose from a variety of jammers available including a cognitive jammer to test anti-jamming technology against. While running, the environment gathers data such as BER (bit error rate) to evaluate system performance.**

***Index Terms*—HF band, Follow-on Jammer, Intelligent Jammer, Reinforcement Learning**

I. Introduction

With the development of new jamming techniques the need for new anti-jamming methods have become necessary to minimize the jamming impact on wireless communications. The issue of jamming has been exacerbated as machine learning algorithms are being used to increase jamming capabilities. Because jamming is in continuous development, the new anti-jamming techniques are necessary to ensure the good quality of wireless communications. In this paper we present an environment to test anti-jamming techniques against different types of jammers in an HF environment.

This environment simulates the ionosphere effects on a transmitted signal while adding the jamming to it. In this work we present an implementation for two types of jammers, one a follower-jammer and the other an intelligent jammer implemented using deep-Q networks. The idea behind this work is to provide a way to develop better anti-jamming techniques in a simulated environment where the user will have the facility to set the right conditions for the simulation. Some parameters that can be changed are: jamming power, channel SNR (signal-to-noise ratio) and ionosphere conditions. At the same time the user will be able to use any of the two types of jammers mentioned previously. This environment was developed using Simulinks Communications Toolbox and Reinforcement Learning Toolbox. With this we created a transmitter and a receiver with passband modulation to simulate the effects of the ionosphere and jamming attacks on the transmitted signal by calculating the BER at the receiver, and looking at the frequency domain to visualize how the transmitter and jamming signal interact with each other.

II. Background and Problem Formulation

Even though there are already a few useful anti-jamming methods, more advanced jammers appeared which need the development of better anti-jamming techniques. This creates a continuous loop in which Jamming and anti-jamming techniques continue to advance in efforts to defeat the other. For this reason, the development and testing of anti-jamming techniques will be necessary to mitigate the impact that jamming has on the RF band. The goal of this project is to build an easy to use environment for people to develop better anti-jamming technologies in the future.

A. Jammers

There are two big jammer categories: elementary and advanced. For this project we created two types of jammers using the definition given in [5] of a follow-on jammer. According to this paper a follow-on jammer ”hops over all available channels very frequently (thousand times per second) and jams each channel for a short period of time (Mpitziopoulos et al, 2007).” Using this definition we developed a jammer that hops across different channels and jamms each one for a short period of time. The other jammer is an intelligent jammer that uses an actor-critic network to learn the hopping sequence of a hopping transmitter.

B. High Frequency Band

Anti-jamming is most notably needed in the HF band. The HF band is located between 3 and 30 MHz and can provide long distance communications by using skywave propagation. In skywave propagation a signal directed to the sky is reflected back to Earth by bouncing against the ionosphere[7]. Three main problems affecting the HF band are the varying characteristics of the ionosphere through time, the large number of users and jamming. In our simulation model we specifically approach the first two problems.

C. Reinforcement Learning and Actor-Critic Networks

Reinforcement learning is a type of machine learning that learns by trial and error. The agent attempts to learn a set of actions it should take based on observations of its environment to achieve a specific objective. Every time the agent selects an action that assists in it completing its objective, it receives a reward/punishments based on the effectiveness of the chosen action. The agent tries to select the action that will produce the highest reward based on the current observation of the environment. Typically, a reinforcement learning framework is implemented using episodes, which are a set amount of time steps during which the agent must learn the highest-rewarding actions. During an episode, the state of the environment may change, which will influence the actions chosen by the agent.

An actor-critic algorithm is a reinforcement learning method that utilizes outputs from two agents called the actor and critic to derive an optimal policy. The work of the actor is selecting the best action for each state and the critic is in charge of telling the actor how good the action taken was. When the actor chooses an action the critic will generate a Q-value representing the long term reward for the action in that state. The idea of actor-critics is to combine policy-based algorithms and value-based algorithms to get the benefits of both. The advantage of policy-based algorithms is they can work with a continuous set of actions, while their main drawback is their high variance when estimating gradients[13] Also, this method does not take into consideration for future actions. For this reasons policy methods tend to learn slower than value methods. On the other hand, value methods have a lower variance than policy methods, however, they require more computational power. With actor-critics the idea is be able to work with a continuous action space and at the same time reducing learning time.

III. Approach

To test the jammers, we used the transmitter and receiver presented in [1]. As shown in Figure 1, the transmitted signal pass through a Matlab function block containing the stdchan function from Matlab[2]. This function adds the ionosphere effects to the transmitted signal and allows us to change the HF channel conditions. The signal then goes through an AWGN channel block that adds noise to it, and then the jamming signal is added by using the sum block. After adding the environmental effects and the jamming signal, the BER is measured to test the transmitters performance. With this configuration, we have control over multiple parameters such as the carrier frequency, the jamming signal frequency, the environmental conditions and the jamming power.

A. Follow-on Jammer Implementation

In order to make the follow-on jammer moves across frequencies, we used the discrete-time VCO (voltage-controlled oscillator) block from Simulink. This block generates a signal which frequency shift, from its Quiescent frequency parameter is proportional to the input signal in which the input signal is interpreted as a voltage [3]. Its main parameters are the quiescent frequency and input sensitivity. If u is the input to the VCO block, then the frequency of the output signal will be equal to quiescient frequency + u * input sensitivity [4].

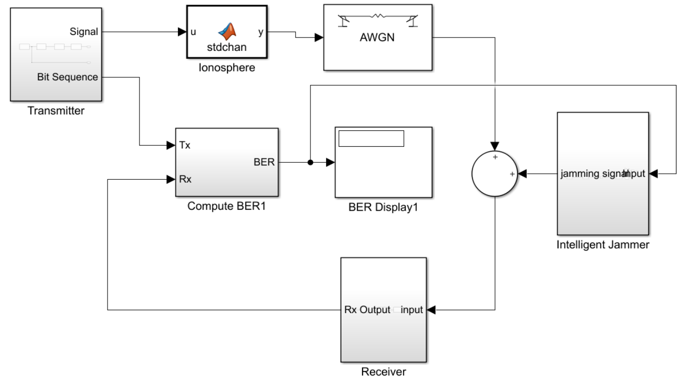

Fig. 1. System model with intelligent jammer connected

The zero holder block controls how much time it will take for the jammer to move from one frequency to the other.

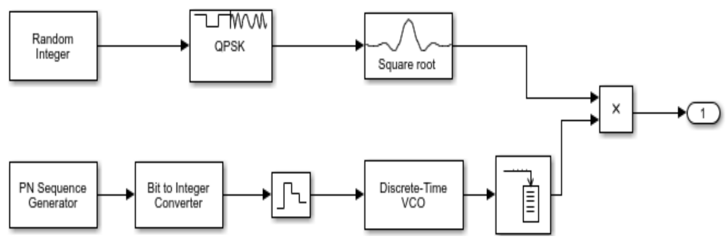

Fig. 2. Follow-on Jammer

B. Intelligent Jammer Implementation

As stated in section IV the intelligent jammer is able to predict the hopping sequence of a hopping transmitter utilizing an actor-critic, as described in section II, for its reinforcement learning algorithm. The learning process is control by the RL Agent block [6], which takes the fast Fourier transform of the transmitted signal as the observation and gives a reward depending on the amount of bit errors produced. Every time there is a bit error at the receiver, a positive reward will be awarded to the agent block. If no bit errors occur, a negative reward will be awarded to the agent. Similar to the follow-on jammer, the discrete-time VCO block is used to change the frequency of the jamming signal,however, now the frequency of the output signal is now controlled by the RL Agent block.

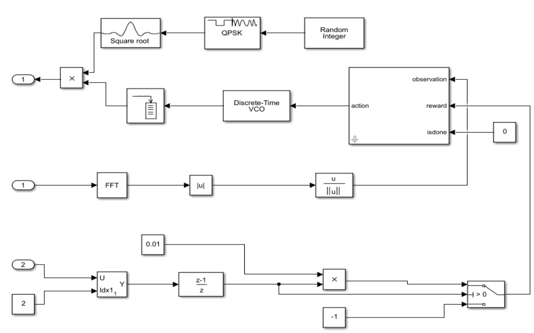

Fig. 3. Intelligent Jammer

IV. EXPERIMENTAL RESULTS

The jammers were tested agaisnt a hopping transmitter by measuring the BER produced by each one. The intelligent jammer trained for 320 episodes as shown in Figure 7. A comparison of the BER produced by both jammers can be seen in Figure 4. It is evident the intelligent jammer performs better than the follow-on jammer by causing a constant BER around 46% while the follow-on jammer caused a BER of 5%. The same test was made with the transmitter hopping at double the velocity. The performance for this simulation can be seen in Figure 5. Same as before, the intelligent jammer caused a 46% BER, however, the BER for the follow-on jammer was a little less than 5%. Another test with the intelligent jammer was made, this time trained for 180 episodes. As can been seen on Figure ... the BER time decrease from 46% to around 43%.

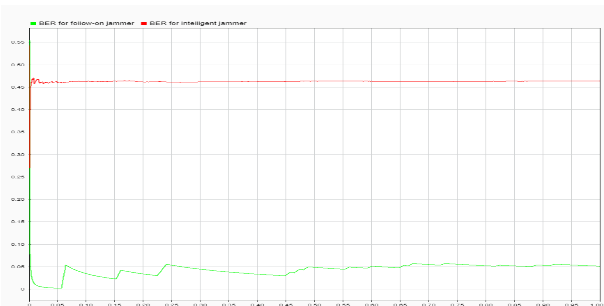

Fig. 4. BER for follow-on jammer and intelligent jammer

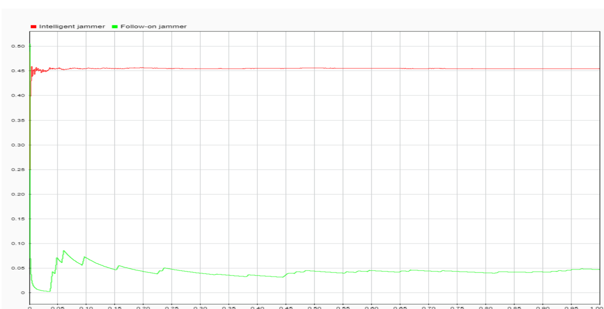

Fig. 5. BER for follow-on jammer and intelligent jammer with transmitter hopping at double the speed.

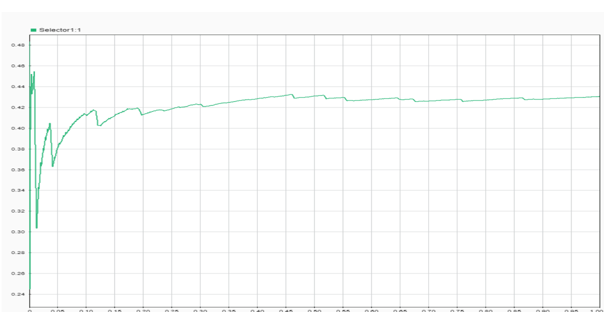

Fig. 6. BER for follow-on jammer trained for 180 episodes

The behavior of the intelligent jammer and the transmitter can be seen in Figures 9 and 10 in which Figure 9 is for the jammer trained for 180 episodes and Figure 10 is for the jammer trained for 320 episodes.

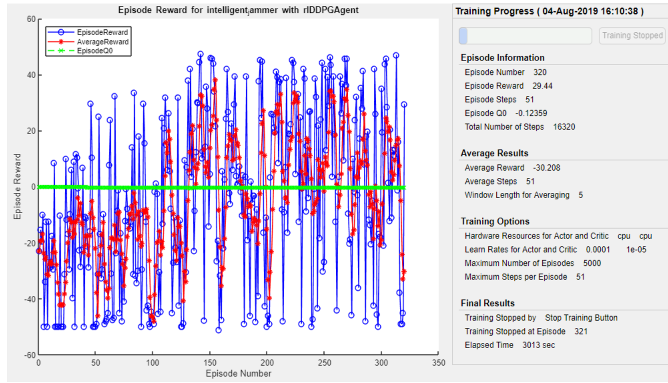

Fig. 7. Graph for 320 episodes of training

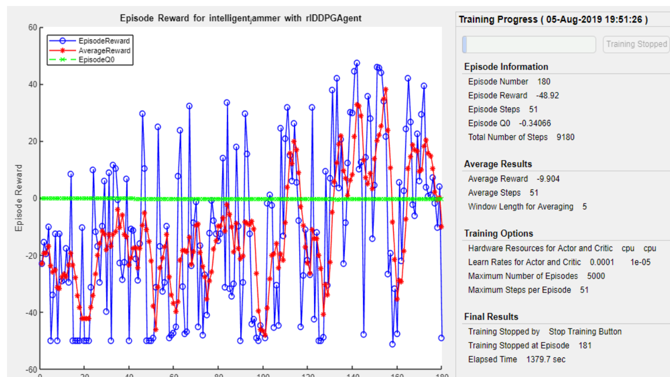

Fig. 8. Graph for 145 episodes of training

V. CONCLUSION AND FUTURE WORK

In this paper we presented an environment for testing anti-jamming in an HF environment and we showed two jammer implementations using Simulink. One future improvement to could be to change the reward function since it is not possible to count the number of bits errors in a real scenario by using a jammer. A way to approach this could be to use a spectrum detection method such as energy detection and reward to the jammer every time the jammer detects the presence of a transmitted signal. Also restricting the frequencies a transmitter can use by adding multiple primary users will make the simulation more realistic. Due to time constraints we were not able to implement as many jammers we wanted. For future development more jammers should be implemented and more capabilities that just tracking should be added to the intelligent jammer.

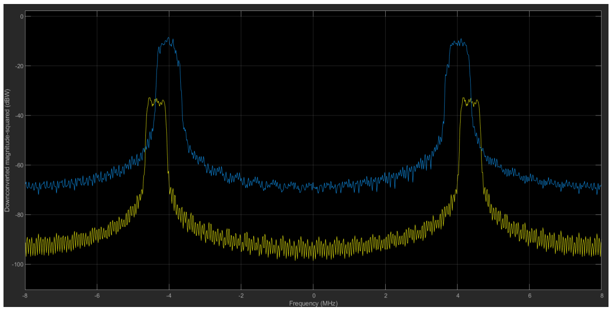

Fig. 9. Frequency of Intelligent Jammer(in blue) and transmitter (in yellow) after 180 training episodes

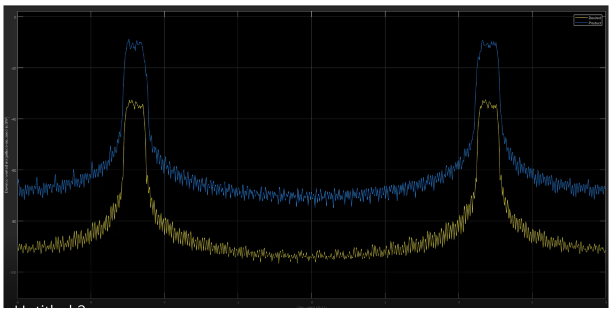

Fig. 10. Frequency of Intelligent Jammer(in blue) and transmitter (in yellow) after 320 training episodes

References

[1] Passband Modulation - MATLAB & Simulink. [Online]. Available: https://www.mathworks.com/help/comm/examples/passband-modulation.html.

[2] stdchan-MATLAB.[Online]. Available: https://www.mathworks.com/help/comm/ref/stdchan.html.

[3] Continuous-Time VCO, Implement voltage-controlled oscillator in discrete time - Simulink. [Online]. Available: https://www.mathworks.com/help/comm/ref/discretetimevco.html.

[4] X. Fan and Z. Tan, Simulink Implementation of Frequency-hopping Communication System and Follower Jamming, 2018 IEEE International Conference on Automation, Electronics and Electrical Engineering (AUTEEE), 2018.

[5] K. Grover, A. Lim, and Q. Yang, Jamming and anti-jamming techniques in wireless networks: a survey, International Journal of Ad Hoc and Ubiquitous Computing, vol. 17, no. 4, p. 197, 2014.

[6] RL Agent, Reinforcement learning agent - Simulink. [Online]. Available: https://www.mathworks.com/help/reinforcement-learning/ref/rlagent.html.

[7] T. Vanninen, T. Linden, M. Raustia, and H. Saarnisaari, Cognitive HF New Perspectives to Use the High Frequency Band, Proceedings of the 9th International Conference on Cognitive Radio Oriented Wireless Networks, 2014.

[8] R. Badiger, M. Nagaraja, and M. Z. Kurian, Design and Development of Frequency Hopping Spread Spectrum Transmitter, International Journal of Electrical, Electronics and Computer Systems (IJEECS), vol. 2, no. 4, 2014.

[9] Yuxi Li, "DEEP REINFORCEMENT LEARNING: AN OVERVIEW," 26 Nov 2018

[10] I. Grondman, L. Busoniu, G. A. D. Lopes, and R. Babuska, A Survey of Actor-Critic Reinforcement Learning: Standard and Natural Policy Gradients, IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 42, no. 6, pp. 12911307, 2012.

[11] Richard S. Sutton and Andrew G. Barto, "Reinforcement Learning: An Introduction," Second Edition, 2014, 2015

[12] Sergios Karagiannakos, The idea behind Actor-Critics and how A2C and A3C improve them, Sergios Karagiannakos, 17-Nov-2018. [Online]. Available: https://sergioskar.github.io/Actor_critics/.

[13] Konda, V. and Tsitsiklis, J. (2019). "Actor-Critic Algorithms." Papers.nips.cc.

[14] I. Grondman, L. Busoniu, G. A. D. Lopes, and R. Babuska, A Survey of Actor-Critic Reinforcement Learning: Standard and Natural Policy Gradients, IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 42, no. 6, pp. 12911307, 2012.