A Game-Theoretic Approach for Mitigating Jamming Attacks in LPWAN

Md Ashikul Haque
Wayne State University
ashik@wayne.edu

Abusayeed Saifullah
Wayne State University
saifullah@wayne.edu

Abstract

Internet of Things (IoT) applications rely on low-power wide-area networks (LPWANs) for gathering data from widely dispersed devices over long distances. As IoT applications are growing in various domains, LPWANs are getting prone to jamming attacks. Jamming causes excessive packet loss, throughput reduction, long delays, and lower energy efficiency in these networks. Existing anti-jamming work in IoT mainly considers jamming in the communication channel between the gateway/base station and the server. In this paper, we propose to mitigate jamming in LPWANs using a game theoretic approach. To our knowledge, this is the first work on anti-jamming in LPWAN. Game theory can realistically model the interaction between jammer and LPWAN nodes because both try to maximize their utility while minimizing their opponent's. While it has been used for anti-jamming approaches in various wireless networks, they either make simplified assumptions or are not directly extendable to LPWANs due to their devices' severe power constraints. A key characteristic of our approach is that we offload all the computations and communications needed for playing the games to the base station that has more power and energy, thereby preserving the energies of the LPWAN devices. The proposed game theoretic approach is designed considering SNOW (Sensor Network Over White spaces), an LPWAN architecture to support scalable wide-area IoT over the TV white spaces. We develop the game based on frequency hopping and transmission power adjustment as anti-jamming actions. Finally, we evaluate the effectiveness of our jamming mitigation technique through both physical experiments and NS-3 simulations. The results show that our technique mitigates jamming by improving packet reception rate, throughput, and energy consumption per packet up to 31.83, 30.77, and 34 times, respectively.

Categories and Subject Descriptors

I.7.3 [**Security and privacy**]: Network security—*Mobile and wireless security*

General Terms

Anti-jamming, Design, Experimentation

Keywords

Low-power wide-area network, Stackelberg Game, Reinforcement Learning, Internet of Things

1 Introduction

The Internet of Things (IoT) applications, such as sensing and monitoring, smart agriculture, and smart city, aim to utilize IoT devices to enhance the quality of life, health, and safety of the communities in both rural and urban areas. Due to the growing demand for these applications, the number of IoT devices is increasing rapidly and is expected to reach approximately 29 billion by 2030 [42]. IoT devices are often powered by batteries, dispersed widely (e.g., in thousands) over long distances, and connected to gateways or base stations (BSs) for data gathering using wireless technologies such as low-power wide-area networks (LPWANs) or wireless sensor networks (WSNs). As a result, they are very prone to wireless jamming attacks. *Jamming* is the disruption of communication between a sender and receiver through the intentional transmission of interfering wireless signals on the same channel by one or multiple adversaries. Severe jamming may cause excessive packet loss, throughput reduction, longer transmission (Tx) delays, and lower energy efficiency in any wireless network.

A number of studies have addressed mitigation of jamming in wireless networks [29, 37, 46, 34, 18, 25, 48, 26, 47, 50, 31]. But these studies do not focus on jamming in the IoT networks. Although several works focus on mitigating jamming in different types of IoT networks, none considers jamming in LPWAN [24, 41, 20, 19]. They consider jamming in the communication channel between the access point/BS and the server. However, the communication channel between the IoT devices and the access point can also be jammed in a real-life scenario. Besides, this kind of jamming is more challenging to mitigate, as the IoT nodes are very power-constrained, unlike access points. In this paper, we propose to mitigate jamming in LPWANs that connect numerous IoT devices to gateways/BSs. Only a few existing works have considered jamming in LPWAN [22, 11, 13, 21, 28]. However, they mainly focus on the effect of jamming in these

networks and hardly discuss about jamming mitigation techniques suitable for LPWAN. None of these works proposes any technique that can mitigate jamming in LPWAN. To the best of our knowledge, this is the first work on mitigating jamming in LPWAN.

We propose to mitigate jamming in LPWAN using a game theoretic approach. Game theory can realistically model the interaction between the jammer and the LPWAN nodes because both try to maximize their utility or benefit while minimizing their opponent's. It has been used before for mitigating jamming in various wireless networks [48, 24, 19, 26, 47, 50, 31]. The system model used in [24, 19] to formulate game theory is not extendable to LPWANs. In [31], the authors use spectrum sensing to defend against jamming that can quickly drain the batteries of the LPWAN devices. In [48], the authors use game theory for transmission power to mitigate jamming. However, their game model reaches equilibrium after one action from each player, which is unrealistic. Besides, using only transmission power is insufficient to mitigate jamming in LPWAN as the devices are extremely power-constrained. In [26, 47, 50], channel hopping for cognitive radios relies on simplified assumptions (e.g., the users and the jammers take actions simultaneously). Besides, frequency hopping alone is insufficient for combating jamming [32]. Except [48], the rest of the works consider simultaneous action from the user and jammer, making them impractical.

We propose a multi-fold game-theoretic approach to mitigate jamming in LPWANs, considering their characteristics and the constraints of their devices. Instead of considering simultaneous action, our approach models the games as the leader and follower subgames. Besides, our approach considers playing the game continuously rather than ending after one move. Unlike the single-game approach used in all the existing works, our approach is designed to handle different jamming situations by playing games using multiple parameters. In our approach, all the computations and communications needed for playing these games will be offloaded to the BS that has more power and energy, thereby preserving the energies of the LPWAN devices.

We develop our game theoretic approach for jamming mitigation in *SNOW (Sensor Network Over White spaces)*. SNOW is an LPWAN architecture to support scalable wide-area IoT over the TV white spaces [40, 38, 39, 36, 35]. *White spaces* are the allocated but locally unused TV channels [9, 10]. Compared to the ISM band, they have a much wider, less crowded spectrum in rural and most urban areas, with an abundance in rural areas [14]. SNOW thus avoids the *crowd* in the *limited* ISM band and the *cost* of the licensed band. Compared to cellular LPWANs, SNOW does not need wired infrastructure, making it suitable in rural and urban areas. It is characterized by asynchronous, low power, and massively concurrent communications between numerous nodes and a BS (base station) over long distances, enabling scalable, wide-area IoT in white spaces [40, 38, 39]. With the rapid growth of IoT, LPWANs will suffer from a crowded spectrum due to long range, making it critical to exploit white spaces. Our approach is extendable to other LPWANs.

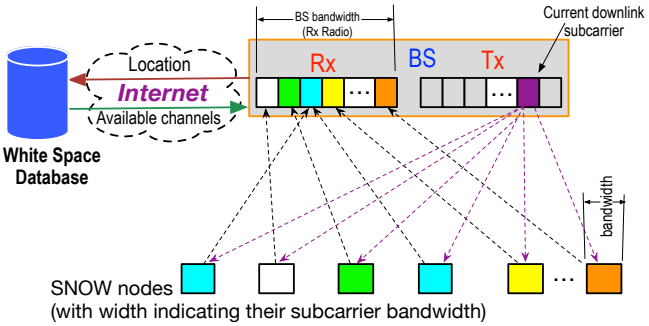


Figure 1: The SNOW architecture.

We design the game theoretic approach for SNOW based on frequency hopping and transmission power as anti-jamming actions. We evaluate the effectiveness of our jamming mitigation technique through both physical experiments and NS-3 [5] simulations. We compare the effectiveness of our proposed game theory approach in SNOW against the current SNOW's link layer protocol (baseline) in the simulation. The results demonstrate that our proposed technique mitigates jamming by improving packet reception rate, throughput, and energy consumption per packet up to 31.83, 30.77, and 34 times, respectively, compared to the baseline.

In the rest of the paper, Section 2 gives an overview of SNOW. Section 3 describes the related works. Section 5 describes the game design, solution, and convergence. Section 6 provides the evaluation results. Finally, Section 7 concludes the paper.

2 An Overview of SNOW

Here we provide a brief overview of the SNOW architecture [38, 39, 40, 36]. Its full description is available in [40]. Due to long transmission range, the nodes in SNOW are directly connected to the BS, forming a star topology as shown in Figure 1. We use **'node'** or **'device'** to indicate a sensor node. The BS knows the locations of the nodes through manual configuration or some existing WSN localization technique [27]. It is line-powered and hence is not powered-constrained. It also has much more computational capability compared to the SNOW nodes/devices. The BS periodically determines white spaces by providing locations of its own and of all other nodes in a cloud-hosted white space database through the Internet. The BS uses wide white space spectrum as a single wide channel consisting of one or more white space channels, which is split into narrowband orthogonal subcarriers, each of equal bandwidth. *Bandwidth* is defined as the width of the spectrum (range of spectrum). Each node has a single half-duplex narrowband radio. It sends/receives on a subcarrier. Like any typical IoT device, the nodes are powered from small batteries as they are usually deployed in various outdoor environments where power sources may not be readily available. Hence, the nodes are energy-constrained, and do not do spectrum sensing or cloud access. As shown in Figure 1, the BS uses two radios – one for only transmission (called ***Tx radio***) and the other for only

reception (called ***Rx radio***) – to facilitate concurrent bidirectional communication.

A key design goal of SNOW is to achieve high scalability by exploiting wide spectrum of white spaces. Hence, its physical layer (PHY) is designed based on a **D**istributed implementation of **OFDM** for multi-user access, called **D-OFDM**. D-OFDM splits a wide spectrum into numerous narrowband orthogonal subcarriers enabling parallel data streams to/from numerous distributed nodes from/to the BS. A subcarrier bandwidth is in kHz (e.g., 50kHz, 100kHz, 200kHz, or so depending on packet size and needed bit rate). Narrower bands have lower bit rate but longer range, and consume less power [16]. The nodes transmit/receive on orthogonal subcarriers, each using one. A subcarrier is modulated using Binary Phase Shift Keying (BPSK) or Amplitude Shift Keying (ASK). If the BS spectrum is split into ν subcarriers, it can receive from ν nodes simultaneously using a single antenna. For downlink transmission, it uses one common subcarrier for every node which is not used for uplink communication [36]. To receive acknowledgement (ACK) of an uplink transmission, a node switches to this downlink subcarrier after making an uplink transmission on its assigned subcarrier. Several subcarriers can be reserved as backup downlink subcarriers.

A SNOW node has communication range of several miles at very low transmission power (e.g., at 0 dBm). Note that the BS can choose to use any part (any bandwidth) of the available white space spectrum based on its need or the availability of white space. The band used by the BS is split into subcarriers and assigned to the nodes for node-BS communication. An added advantage of the SNOW design is that it allows to use fragmented spectrum at the Rx radio of the BS. When we cannot find consecutive white space channels while needing more, the BS may use non-consecutive channels as a single channel. The subcarriers in the unused part of the BS's chosen spectrum will not be used and will be ignored in encoding and decoding.

Currently, the sensor nodes in SNOW use a very simple and lightweight CSMA/CA approach for transmission like the one used in TinyOS [7]. Since each node (non BS) has just a single half-duplex radio, it can be either receiving or transmitting, but not doing both at the same time. The nodes join the BS through a joining process using special subcarriers that are relatively stable.

SNOW has been implemented on USRP (universal software radio peripheral) [2] and TI CC1310 [6] devices. Currently a dual-radio USRP (universal software radio peripheral) [2] device connected to a computer is used as the SNOW BS while the CC1310 devices work as SNOW nodes [36]. An open-source implementation of SNOW is available at [1].

3 Related Work

Many existing works have studied techniques for handling severe interference or jamming for wireless networks [33]. These works mainly rely on spread spectrum techniques, increased Tx power, antenna polarization or directional transmission, and packet fragmentation [29, 37, 46, 34, 18, 25]. Most of the above approaches are tailored for the IEEE 802.15.4 standard [46, 34, 18] only. While Tx power control is a common technique that applies to the LPWANs, using only Tx power to combat jamming is often ineffective since the maximum Tx power of an LPWAN node is quite low (typically $\leq$ 20dBm).

Several jamming mitigation approaches depend on regular message exchanging with a central coordinator node, causing considerable energy consumption and degrading the scalability [29, 37, 25, 18]. JAMMY [44] is an approach for WSN based on TDMA (Time Division Multiple Access) MAC protocol requiring strong time synchronization both in the network and between the network and the jammer. Thus, it may not be suitable for an LPWAN. Jamming mitigation through multipath routing has been studied as well [8, 30], which also may not be applicable to LPWANs as the latter adopt machine to machine (M2M) communication.

Several game theoretic approaches have been proposed for modeling jamming over the past few years, including modeling communication channel [23, 15], wireless networking [12, 49], IoT networks [24, 41, 20, 19], and cognitive radios [17, 26, 47, 50]. These models are based on a simple zero-sum game between the jammer and transmitter-receiver pairs, having hierarchical games (non-cooperative Stackelberg games) or even cooperative games if the receivers can collaborate to achieve the goals. Additionally, they derive the optimal defense strategies based on adversarial conditions using different parameters (e.g., transmission power, hopping, spectrum sensing, and distribution over multiple spectrums).

In [26, 47, 50], channel hopping for cognitive radios relies on simplified assumptions that the users and the jammers take actions simultaneously. As studied in [32], frequency hopping alone is not sufficient for combating jamming. Besides, these approaches cannot be adopted directly in LPWANs. A recent work has studied a game theoretic approach for handling the jamming attack on control channels in cognitive radio networks [31]. Adopting that policy for LPWAN (SNOW) will require spectrum sensing to determine the spectrum at each device, which is highly energy consuming and can quickly drain the batteries of the devices. Mitigating jamming between the access point and the Internet has been studied in [48, 24, 19]. However, it does not provide any solution for jamming on the nodes. In [48], the game considers only one move from each player, which is not applicable in real-world.

In this paper, we propose a complex and practical game-theoretic approach instead of a simple game-theoretic approach to mitigate jamming on LPWAN nodes. Our designed game is suitable for LPWAN devices having low computation power and battery life. The BS computes and plays these games on behalf of the nodes. When the game reach convergence, the BS establish the LPWAN (SNOW) network. Unlike the existing work [26, 47, 50], the BS and jammer need not take actions simultaneously in our model. Instead, they follow a leader and follower game model where the jammer keeps playing the game against the BS as a follower, thereby making our approach a practical and effective choice.

4 System and Attack Model

In this section, we describe our system model as well as the attack model used by the jammer.

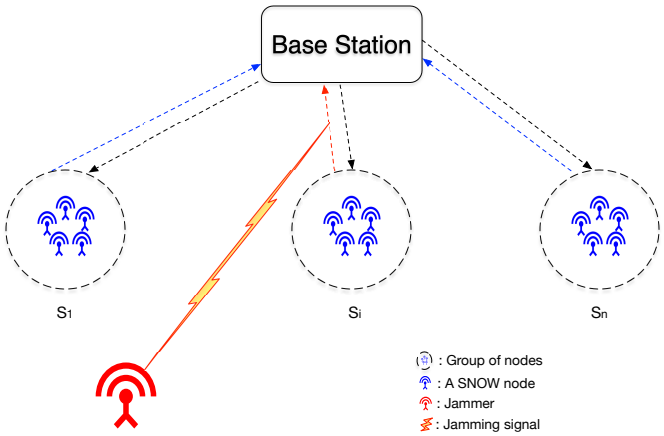


Figure 2: The jamming attack model in SNOW.

We consider a jamming attack scenario in an LPWAN based on SNOW and develop our anti-jamming techniques. Specifically, we consider a SNOW network with an Internet-connected BS and a set of SNOW nodes/devices that directly communicate with the BS. In addition, there is a jammer that will intentionally try to disrupt the node to BS communications in the SNOW network.

Note that the SNOW BS gets the list of white space channels through a cloud-hosted database by accessing the Internet. Let the set of k available white space channels be denoted by $\mathcal{W} = \{w_1, w_2, \cdots, w_k\}$. These channels have center frequencies $\mathcal{F} = \{f_1, f_2, \cdots, f_k\}$ where f_i is the center frequency of white space channel w_i. We consider n groups (subsets) of SNOW nodes $\mathcal{S} = \{S_1, S_2, \cdots, S_n\}$, as shown in Figure 2. Each group (set) of nodes S_i has up to max_i number of SNOW nodes, and all nodes of a group S_i use a single white space channel w_i. Namely, each node in S_i is assigned a subcarrier from the white space channel w_i. From the available white spaces, suppose the BS (i.e., the SNOW network) uses the first n channels from set $\mathcal{W}$ denoted by $\mathcal{W}_{\mathcal{U}} = \{w_1, w_2, \cdots, w_n\}$ where $n \leq k$. The BS keeps the list of the remaining white space channels (i.e., available but unused white space channels) $\mathcal{W}_{\mathcal{A}} = \{w_{n+1}, w_{n+2}, \cdots, w_k\}$. Every node of a set s_i has its uplink subcarrier assigned by the BS.

The jammer tries to jam the uplink subcarriers (used for node to BS communications) of a set of nodes S_i, as shown in Figure 2. Unlike uplink subcarriers used for transmissions by the SNOW nodes that are characterized by severe power-constraints, the downlink subcarriers are used for transmission by the BS that does not have power constraints. Therefore, the jammer does not waste its energy by jamming downlink subcarriers. Note that, in this paper, we consider there is one jammer trying to jam the SNOW nodes $\mathcal{S}$. It cannot jam the uplink subcarriers of more than one white space channel w_i at a time. Hence, the jammer can choose to jam any group of nodes from $\mathcal{S}$, but only one group of nodes S_i can be jammed at a time. In doing so, the jammer is able to interrupt all the uplink communications between the group of SNOW nodes S_i and the BS.

The jammer is reactive and can change its transmission power and frequency by sensing the changed transmission power and frequency used by the SNOW nodes. We consider that the jammer has an energy budget for jamming our SNOW network. This is a practical consideration as the cost of jamming can outweigh the gain otherwise. Hence, it needs to use its energy effectively and efficiently. When the jammer reaches its energy budget, it stops jamming the SNOW network.

5 Proposed Game-Theoretic Approach for Jamming Mitigation

In this section, we develop our game theoretic approach for anti-jamming in SNOW. Note that game theory is a realistic choice for this as it can model the interaction between the jammer and the LPWAN nodes better as both parties try to maximize their utility or benefit while minimizing the same of their opponent's. While it has been used before to develop anti-jamming approaches in various wireless networks [48, 24, 19, 26, 47, 50, 31], they are either based on simplified assumptions or are unsuitable for LPWANs because of their unique characteristics and severe power-constraints of the devices. Therefore, we have developed a new game theoretic approach for LPWAN, particularly considering the unique characteristics and constraints of SNOW.

In this game, the players are the SNOW nodes and the jammer. As mentioned in Section 4, the jammer is jamming the uplink communication between the nodes and the BS. The nodes try to maximize their total network throughput, while the jammer tries to minimize their throughput. In our case, the SNOW nodes are in a disadvantageous situation compared to the jammer, as the nodes have significantly less transmission power, battery life, and computation power than the jammer. Besides, playing the game severely affects the SNOW nodes' battery life, as it adds a computation overhead before each transmission. In such an adverse matchup, we have to design the game to ensure there is no computation overhead for the nodes and they can keep transmitting important sensor data successfully using low transmission power. Therefore, we propose to offload all the computations and communications for the game to the BS. After detecting jamming on the nodes' uplink subcarrier, the BS plays the game on the nodes' behalf until the convergence of the game and determines the nodes' action. Then it sends downlink transmissions containing information about the nodes' actions. The BS can use much higher TX power to combat jamming in downlink channel. Hence, in this scenario, the downlink is considered not to be jammed. The nodes decode the control message and take action prescribed by the BS.

5.1 Game Design

There are many complexities in real-world jamming. Thus, instead of simplifying our game by using just one parameter (i.e., transmission power/ channel hopping), we propose to change both of these parameters together to enhance the accuracy of the game. In the game, the total utility depends on all the parameters (i.e., signal-to-noise ratio, packet

reception rate, expected channel stealth, etc.), reflecting the impact of all of them and making the game complex. Furthermore, to make the game more realistic, we design our game as a **Stackelberg game** instead of a Nash game. In Nash game, there is an assumption that both players will take action simultaneously. However, this aligns differently from the real-life scenario of jamming, where a player can take action after the other player has taken action. Thus, a Stackelberg game is more suitable here as there is a leader and a follower. For our game, the nodes are the leader, and the jammer is the follower. The jammer can observe the action taken by the leader and choose its action accordingly. Even though it makes the game more challenging for the nodes, it is more realistic compared to Nash for this scenario.

5.1.1 Transmission power

Whether a received packet from the nodes of the jammed group S_i will be successfully decoded or not depends largely but not only on the signal-to-noise ratio (SNR) at the BS. Now, the nodes in group S_i want to increase the packet transmission rate and throughput at the BS and decrease energy consumption for packet transmission. If the BS can decode the sent packets successfully, the nodes in group S_i will achieve a better packet delivery rate and throughput. This ensures that no retransmission is needed for that packet, which can help decrease energy consumption. Hence, the utility of this game heavily depends on the value of SNR for both players. However, we must also consider the transmission power used for the battery-powered nodes. Because even though higher transmission power increases SNR, resulting in better packet transmission rate and throughput, it severely affects the nodes' battery life. Therefore, the utility of the nodes needs to have a penalty for using higher transmission power. We design the utility $U_i^P(t)$ for transmission power $P_i(t)$ of the nodes in the group S_i at step t as below.

$$U_i^P(t) = \text{SNR}(P_i(t), P_J(t)) - P_i(t)$$

To further improve the utility, we incorporate the *expected Packet Reception Rate (PRR)* to account for the uncertainties in the complex communication channels. We achieve this by introducing a function $\text{PRR}(P_i(t), P_J(t))$ that returns the expected PRR for node transmission power $P_i(t)$ and jamming transmission power $P_J(t)$ at step t by using the probabilistic model obtained from previous (t-1) steps. The modified utility $U_i^P(t)$ of the nodes in group S_i at step t is as below.

$$\begin{aligned} U_i^P(t) =& \text{SNR}(P_i(t), P_J(t)) - P_i(t) \\ &+ \text{PRR}(P_i(t), P_J(t)) \end{aligned}$$

As the nodes in group S_i aim to maximize the utility, the optimal transmission power $P_i^*(t)$ at step t is as follows.

$$\begin{aligned} P_i^*(t) &= \underset{P_i \le \bar{P}}{\arg\max}\ U_i^P(t) \\ &= \underset{P_i \le \bar{P}}{\arg\max}\ \Big(k_1 \times \text{SNR}(P_i(t), P_J(t)) - k_2 \times P_i(t) \\ &\qquad + k_3 \times \text{PRR}(P_i(t), P_J(t))\Big) \end{aligned} \tag{1}$$

where k_1, k_2, k_3 are the weights and $\bar{P}$ is the maximum transmission power that a node can use.

For the jammer, a higher SNR of the nodes in the group S_i results in a higher penalty. The jammer has a power budget, so it imposes a penalty if the higher transmission power is used for jamming. Furthermore, the jammer can use a function $\text{PRR}'(P_i(t), P_J(t))$ that returns the estimated expected PRR of the nodes in the group S_i for jamming transmission power $P_J(t)$ at step t. Therefore, the utility of the jammer $U_J^P(t)$ at step t can be expressed as follows.

$$\begin{aligned} U_J^P(t) =& -\text{SNR}(P_i(t), P_J(t)) - P_J(t) \\ &- \text{PRR}'(P_i(t), P_J(t)) \end{aligned}$$

As the jammer aims to maximize the utility, the optimal transmission power $P_J^*(t)$ at step t is given by:

$$\begin{aligned} P_J^*(t) &= \underset{P_J \le \bar{P}_J}{\arg\max}\ U_J^P(t) \\ &= \underset{P_J \le \bar{P}_J}{\arg\max}\ \Big(-b_1 \times \text{SNR}(P_i(t), P_J(t)) - b_2 \times P_J(t) \\ &\qquad - b_3 \times \text{PRR}'(P_i(t), P_J(t))\Big) \end{aligned} \tag{2}$$

where b_1, b_2, b_3 are the weights and $\bar{P}_J$ is the maximum transmission power the jammer can use.

Even though eq. (1) seems optimal, it is not considered the optimal solution for a Stackelberg game. In the Stackelberg game, the leader's optimal action considers the follower's optimal action. Thus, considering the optimal transmission power of the jammer $P_J^*(t)$, we rewrite eq. (1) as follows, which is the Stackelberg optimal transmission power for nodes in group S_i at step t.

$$\begin{aligned} P_i^*(t) = \underset{P_i \le \bar{P}}{\arg\max}\ \Big(&k_1 \times \text{SNR}(P_i(t), P_J^*(t)) \\ &- k_2 \times P_i(t) + k_3 \times \text{PRR}(P_i(t), P_J^*(t))\Big) \end{aligned} \tag{3}$$

Therefore, the tuple $(P_i^*(t), P_J^*(t))$ corresponding to the eqs. (1) and (2) is the Stackelberg equilibrium for the transmission power.

5.1.2 Frequency Hopping

Along with transmission power, the nodes in group S_i adopt frequency hopping. Frequency hopping refers to changing the white space channel for all the nodes in group S_i. Frequency hopping combined with transmission power can be a powerful tool to mitigate jamming. However, before

hopping to a white space channel at any step t, the group of nodes needs to know the expected search time of the jammer for that white space channel. For now, let us assume we have a function $h(f_i(t))$ that returns the expected utility using the probabilistic model based on search time of the jammer, obtained from previous (t-1) steps. Thus, the utility function $U_i^F(t)$ for the nodes of group S_i at step t for hopping to a white space channel with frequency $f_i(t)$ is as follows.

$$U_i^F(t) = h(f_i(t))$$

As the nodes in group S_i aim to maximize the utility, the optimal white space channel at step t with frequency $f_i^*(t)$ is as follows.

$$\begin{aligned} f_i^*(t) &= \underset{f_{min} \leq f_i \leq f_{max}}{\arg\max} \quad U_i^F(t) \\ &= \underset{f_{min} \leq f_i \leq f_{max}}{\arg\max} \quad k_4 \times h(f_i(t)) \end{aligned} \tag{4}$$

where k_4 is the weight, f_{min} and f_{max} are the minimum and maximum center frequencies for the TV channels.

While designing the utility function of the jammer, we have to account for the expected count of hops in step t besides the expected search time of the jammer for that white space channel. To incorporate these, we introduce an additional function $c(f_J(t))$ that returns the expected utility using the probabilistic model based on the count of hops for the white space channel with frequency $f_J(t)$, obtained from the previous $(t-1)$ steps. Thus, below is the utility function $U_J^F(t)$ for the jammer at step t for hopping to a white space channel with frequency $f_J(t)$.

$$U_J^F(t) = -h(f_J(t)) - c(f_J(t))$$

As the jammer's goal is to maximize the utility, the optimal white space channel at step t with frequency $f_J^*(t)$ is as follows.

$$\begin{aligned} f_J^*(t) &= \underset{f_{min} \leq f_J \leq f_{max}}{\arg\max} \quad U_J^F(t) \\ &= \underset{f_{min} \leq f_J \leq f_{max}}{\arg\max} \quad \Big(-b_4 \times h(f_J(t)) - b_5 \times c(f_J(t))\Big) \end{aligned} \tag{5}$$

where b_4, b_5 are the weights.

Akin to transmission power, even though eq. (4) seems optimal, it is not considered the optimal solution for a Stackelberg game. In the Stackelberg game, the leader's optimal action considers the follower's optimal action. Thus, considering the optimal white space channel frequency of the jammer $f_J^*(t)$, we modify eq. (4) as below, which is the Stackelberg optimal white space channel frequency for hopping for nodes in group S_i at step t.

$$f_i^*(t) = \underset{f_{min} \leq f_i \leq f_{max}}{\arg\max} \quad k_4 \times h(f_i(t), f_J^*(t)) \tag{6}$$

Therefore, the tuple $(f_i^*(t), f_J^*(t))$ corresponding to the eqs. (4) and (5) is the Stackelberg equilibrium for the white space channel frequency.

5.1.3 *Combining Transmission power and Frequency Hopping*

Even though we designed the Stackelberg equilibrium separately for the transmission power and frequency hopping, we can combine the equilibriums to obtain the final Stackelberg equilibrium for our game. We can get the optimal formulation for the nodes of the group S_i for step t by combining the eqs. (3) and (6), where G_i is our game function for nodes of group S_i.

$$\begin{aligned} G_i(P_i^*(t), f_i^*(t)) = \underset{\substack{P_i \leq \bar{P} \\ f_{min} \leq f_i \leq f_{max}}}{\arg\max} \Big(& k_1 \times \mathrm{SNR}(P_i(t), P_J^*(t)) \\ & - k_2 \times P_i(t) \\ & + k_3 \times \mathrm{PRR}(P_i(t), P_J^*(t)) \\ & + k_4 \times h_i(f_i(t), f_J^*(t)) \Big) \end{aligned}$$

Similarly, by combining the eqs. (2) and (5), we can get the game function for the jammer as below.

$$\begin{aligned} G_J(P_J^*(t), f_J^*(t)) = \underset{\substack{P_J \leq \bar{P_J} \\ f_{min} \leq f_J \leq f_{max}}}{\arg\max} \Big(& -b_1 \times \mathrm{SNR}(P_i(t), P_J(t)) \\ & - b_2 \times P_J(t) \\ & - b_3 \times \mathrm{PRR}'(P_i(t), P_J(t)) \\ & - b_4 \times h_J(f_J(t)) \\ & - b_5 \times c_J(f_J(t)) \Big) \end{aligned}$$

Note that even though the game function for the nodes of S_i depends on optimal transmission power and white space channel frequency of the jammer for step t, the jammer's game function does not incorporate such dependency as it takes its action after observing the optimal action taken by the nodes of S_i. Furthermore, the optimal transmission power and white space channel frequency can be used by every node of group S_i, when the game reaches convergence.

5.2 Solution of the Game

As our proposed game function is complex by incorporating a lot of real-world scenarios and can not be mathematically modeled beforehand, applying the traditional equation-solving approach is not realistic. Many functions (i.e., expected channel search time, expected packet reception rate, expected number of hops) related to the game depend on the quality of communication in the real world and can not be mathematically modeled accurately beforehand. Thus, we propose to solve this game function incorporating a learning-based approach through using *novel node agent and jammer agent* based on Reinforcement Learning (RL) [43]. Specifically, we propose to use **Non-cooperative Multi-Agent Reinforcement Learning (NMARL)** to solve our game. Reinforcement Learning is a sub-field of machine learning that focuses on learning by rewarding and penalizing any executed action by the agents in a complex environment. The

main advantage of reinforcement learning lies in modeling complex real-world scenarios through interaction with the environment by trial and error.

To leverage the advantages of reinforcement learning to solve our game, we use the BS (on behalf of the nodes) and the jammer as agents. According to our game functions, both of the agents are non-cooperative, meaning their actions try to focus on minimizing the other's reward while maximizing its own. To incorporate our game utility for both the nodes and the jammer, we propose to use their utilities as reward and penalty in the reinforcement learning framework. To accommodate all the complex functions (i.e., expected channel search time, expected packet reception rate, the expected number of hops), we use complex communication channels in the real-world as the environment for our NMARL.

Understanding the use of NMARL. NMARL is suitable for our game, as the agents interact with the complex communication channel that is very difficult to model without prior interactions. Furthermore, NMARL can give a very good quality solution for the optimization problem formulated by the game-theoretic approach. The node agent takes action that maximizes its total utility while predicting the next optimal action of the jammer agent. The jammer agent observes the action taken by the node agent and decides its optimal action for that step. This is exactly the same as the Stackelberg game.

Moreover, the agents affect each other's reward through their actions while not sharing any information and trying to reduce the reward of the other agent. However, it is difficult for the low-powered and computationally-constraint SNOW nodes to interact with the environment and execute action against the other agent. Thus, we propose to use the BS as the node agent. The BS interacts with the environment and executes the action on behalf of the nodes during the training process. Specifically, during training the BS uses its Tx radio to transmit on behalf of SNOW nodes while receiving through the Rx radio mimicking the SNOW nodes' action.

Translating the game functions to NMARL. Some game functions (i.e., expected channel search time, expected packet reception rate, the expected number of hops) can become easier to predict while using the NMARL. We propose quantifying the channel search time, packet reception rate, and the number of hops for each step through packet transmission by both the jammer and BS. During the training, the expected value of these parameters can be obtained after each action executed by the agents at each step t. All of these values are reflected through the agents' reward value (i.e., total utility) for step t. The reward value for all the actions can be stored in a Q-table [45]. Note that the reward value is obtained using the agents' utility derived in the Subsection 5.1. Even though using NMARL we can get a very good quality solution for our complex game function, it does not guarantee an optimal result.

Learning process. The agents are trained for all possible actions by interacting with each other through the environment (complex communication channel). The reward values that reflect the game utility for all the actions can be stored in a Q-table as part of Q-learning [45]. Note that both agents have separate Q-table, and it is not shared. It is not necessary

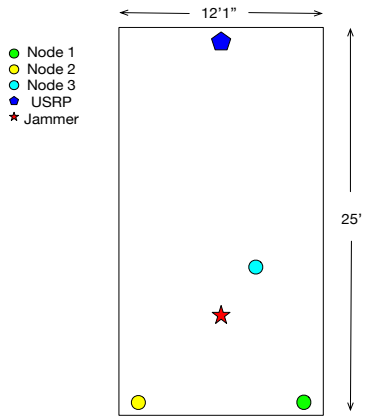


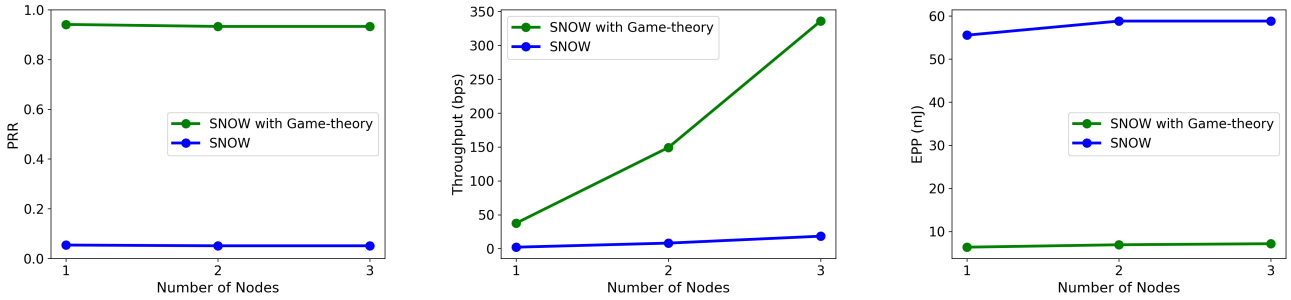
Figure 3: Experimental setup.

that this training has to happen against a real jammer. The BS (node agent) can be trained against a mock jammer and the vice versa. Later, the BS can use the trained data against a real jammer. After training, the BS takes a new action only if an action can yield higher utility than the current one. Therefore, it is important to explore all the actions in the action space and the combination of the actions against each other for both agents during training. This ensures proper modeling of the communication channel and the other agent's action.

Proof of Convergence. At every step of the game, the agents take an action only when the action can yield higher utility than the current one. This implies that our game utility function is monotonically increasing, which means after finite time steps there will be no other action in the finite action space that can yield higher utility than the current one. In that scenario, neither the node agent nor the jammer agent has any incentive to take an action, which defines the convergence of the game. When the game reaches convergence, the BS assigns the white space channel for the SNOW nodes of group S_i. Note that before convergence, the BS plays the game against the jammer on behalf of the SNOW nodes of group S_i.

6 Evaluation

We first train the RL model using a mock jammer in NS-3 [5] for 18 hours to evaluate our proposed game-theoretic approach for mitigating jamming in SNOW. After training the model, it is used in both Experiment and Simulation. From the discussion in Section 3, it is evident that the existing jamming mitigation techniques cannot be applied to LPWANs (i.e., SNOW [38, 39, 40], LoRA [4]). Therefore, we compare our anti-jamming technique with the current SNOW protocol as a baseline in both the experiment and simulation. Furthermore, we compare our proposed approach to the separate usage of transmission power and frequency hopping in simulation, as it can demonstrate how much improvement our combined approach adds to communication quality.



(a) Packet Reception Rate (b) Throughput (c) Energy per packet

Figure 4: Experimental result.

6.1 Experiment

We perform a small-scale experiment using an indoor setup to evaluate our approach. We load our previously trained RL model in the BS for our experiment.

6.1.1 Setup

As depicted in Figure 3, we implement our experiment in an indoor setup. We use GNU radio [3] and two Universal Software Radio Peripheral (USRP) B200 devices [2] to implement the BS and jammer. Along with the game implementation, we set the jammer to transmit a jamming packet whenever it senses any signal on the TV white space channel. The jammer also uses the same trained RL model as the BS. As SNOW nodes, we use TI CC1310 [6], a commercial off-the-shelf (COTS) device. A SNOW node's transmission interval, packet size, and total packet transmission are set to 1 minute, 40 bytes, and 1000, respectively. We assign 98 kHz bandwidth for each node and set the maximum retransmission number to 2. The nodes start to transmit packets after the game convergence in the BS. Maximum transmission power for the SNOW node is set to 15 dBm, while the jammer has maximum transmission power of 25 dBm.

6.1.2 Experimental Results

For evaluation, we use three metrics (i.e., Packet Reception Rate (PRR), Throughput, and Energy Per Packet (EPP)).

- PRR is calculated as $\text{PRR} = \frac{\text{Total Received Packet}}{\text{Total Transmitted Packet}}$.
- Throughput is calculated as $\text{Throughput} = \frac{\text{Total Received Packet at the BS} \times \text{Packet Size}}{\text{Total Time}}$.
- EPP is calculated as $\text{EPP} = \frac{\text{Total Energy for Transmission}}{\text{Total Received Packet}}$.

Note that Total Energy for Transmission includes the additional energy used for retransmission. Thus, lower EPP indicates better performance, while higher PRR and throughput show better performance.

We vary the number of nodes from 1 to 3 for our experiment. As depicted in Figure 4(a), irrespective of the number of nodes, our approach achieves a high PRR of 94%, which is significantly better than vanilla SNOW (the baseline) that has a PRR of only 5.4%. Moreover, our approach demonstrates substantially improved throughput performance and achieves a throughput of up to 335.88 bps compared to 18.36 bps of the baseline as depicted in Figure 4(b). Finally, it outperforms the baseline vastly by decreasing the energy consumption from 58.82 mJ to 6.37 mJ per packet as depicted in Figure 4(c). In terms of all the metrics, our approach achieves a very good quality of communication under jamming.

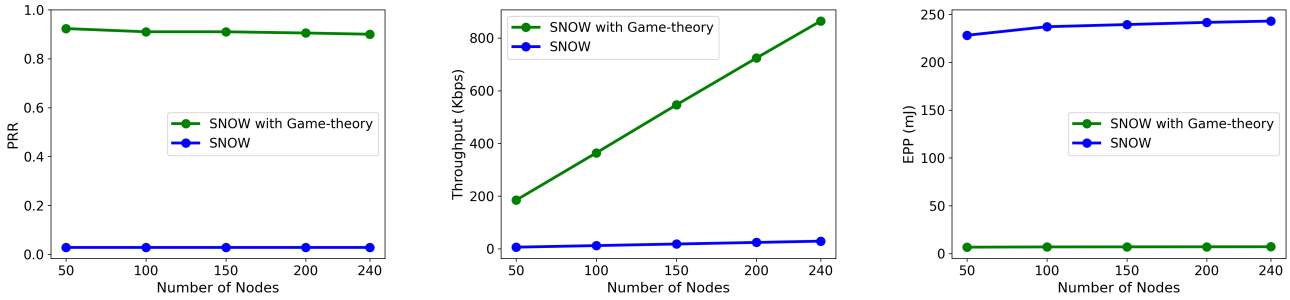
6.2 Simulation

We use NS-3 [5] for large-scale evaluation. In the simulation environment, the nodes are situated within a 690-meter radius of the BS, while the jammer is placed 700 meters from the BS. Moreover, all the nodes and the jammer are stationary in this simulation. The packet size and transmission interval are set to 40 bytes and 1 sec in the simulation. Moreover, the SNOW network uses a single white space channel for this simulation, which ensures all the nodes are affected by the jammer. We run simulation for 120 minutes for each setup (i.e., varying number of nodes, varying number of retries). We use the same trained RL model for all setups for consistency. During the simulation, the game reached convergence within 10-15 minutes, and the BS informed the jammed nodes of their new white space channel and transmission power. Thus, the jammed nodes started transmission after 10-15 minutes.

6.2.1 Performance under Varying Number of Nodes

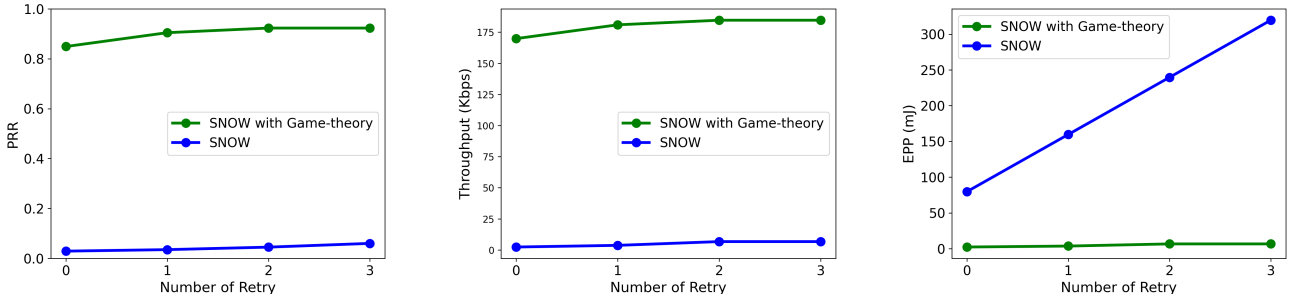
In this simulation, we vary the number of nodes from 50 to 240 while the maximum retransmission number is 2. Increasing the number of nodes too much in a white space channel can decrease the bandwidth (i.e., subcarrier width) significantly for each node. In the simulation, we noticed that we could increase the number of nodes to 240 without compromising the performance much when there is no jammer. Thus, we choose 240 as the highest number of nodes to evaluate our approach in the presence of a jammer.

As depicted in Figure 5(a), even when the number of nodes increases, the PRR remains almost the same. It drops very slightly for higher number of nodes, which occurs due to lower bandwidth. The throughput increases almost linearly as shown in Figure 5(b) with the increasing number of nodes, and the reason is the increasing number of transmitted packets by a higher number of nodes while the PRR remains similar to the lower number of nodes. The EPP increases slightly due to the higher number of packet retransmission for the increasing number of nodes. Even though the PRR



(a) Packet Reception Rate (b) Throughput (c) Energy per packet

Figure 5: Performance under varying number of nodes (in NS-3).



(a) Packet Reception Rate (b) Throughput (c) Energy per packet

Figure 6: Performance under varying number of retransmissions (in NS-3).

is almost the same, it is achieved through more retransmissions than the lower number of nodes. Overall, our approach outperforms the baseline by 31.83, 30.77, and 34 times for PRR, throughput, and EPP, respectively.

6.2.2 *Performance under Varying Number of Retransmissions*

In this setup, we vary the maximum number of retransmissions from 0 to 3. We notice that increasing the number of retransmissions does not improve performance beyond two retransmissions in the presence of the jammer.

Figure 6 shows the impact of the varying number of retransmissions. From Figure 6(a), it is eminent that without any retransmission, the PRR becomes lower. However, when the number of retransmission is 2 or 3, the PRR is almost the same. The throughput is analogous to the PRR for the varying number of retransmissions. Even though more retransmissions improve communication quality, they can significantly increase energy consumption, as depicted in Figure 6(c). Even though the increase seems negligible compared to the baseline, it can impact power-constraint SNOW nodes' battery life. The SNOW without any game-theory can improve PRR and throughput by at most 1% by retransmitting the packets. The performance improvement is very trivial for PRR and throughput. Furthermore, the higher number of retransmissions causes very high energy consumption resulting in very high EPP. Compared to the baseline, our game-theoretic approach performs significantly better for all the metrics.

6.2.3 *Performance Comparison with Non-Combined Approaches*

In this simulation, we compare our combined game-theoretic approach with the non-combined game-theoretic approach. We train two new RL models using transmission power and frequency hopping separately. Thus, one RL model plays the game using only transmission power, while the other uses only frequency hopping. We use these two models in the simulation and compare their results with our combined game-theoretic approach. For this simulation, we use 50 nodes and two maximum numbers of retransmissions. All other variables (i.e., simulation time, distance) remain the same as other setups.

As depicted in Figure 7, the game-theoretic approach that combines both the transmission power and frequency hopping is outperforming isolated transmission power and frequency hopping significantly for all the metrics. This result demonstrates that our proposed game-theoretic approach can outperform basic SNOW protocol as well as the game-theoretic approaches with one knob. From Figure 7, it is noticeable that even though the isolated knobs perform inadequately compared to our proposed approach, they still outperform the basic SNOW protocol by a large margin.

Among the isolated knobs, the transmission power

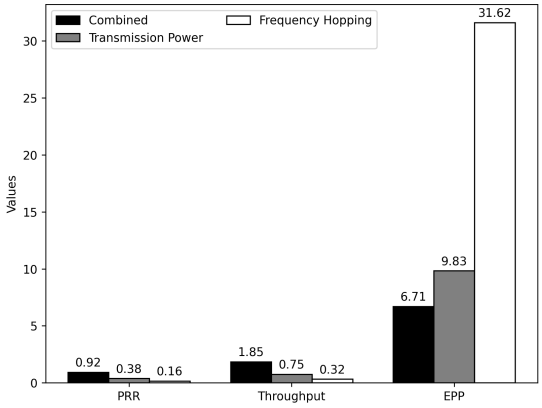


Figure 7: Comparison with non-combined approaches (in NS-3).

evinces its superiority in all the metrics and time of convergence. During simulation, we noticed that the transmission power game could converge faster than the frequency hopping game. In fact, during multiple 120 minutes of simulation duration, the frequency hopping game never reached the convergence. Even though the transmission power game converges, it takes more than 1 hour to reach convergence compared to 10-15 minutes of the combined approach. As for the metrics, the transmission power game achieves more than double PRR and throughput while having a significantly better EPP.

All the results demonstrate that our proposed game-theoretic approach for mitigating jamming in SNOW improves the network performance significantly.

7 Conclusion

In this paper, we have proposed a game-theoretic approach for mitigating jamming attacks in LPWANs. Today, IoT applications rely on LPWAN technologies for gathering data from widely dispersed devices over long distances. As IoT applications are rapidly growing in various domains, including smart cities, smart agriculture, healthcare, and community safety, LPWANs are getting prone to jamming attacks. Jamming may cause excessive packet loss, throughput reduction, longer delays, and lower energy efficiency in these networks.

We have developed the proposed game theoretic approach considering SNOW, an LPWAN architecture to support scalable wide-area IoT over the TV white spaces. While game theory has been used for anti-jamming approaches in various wireless networks, they either make simplified assumptions or are not directly extendable to LPWANs due to their devices' severe power constraints. A key characteristic of our approach is that we offload all the computations and communications needed for playing the games to the base station that has more power and energy, thereby preserving the energies of the LPWAN devices. We have developed the game based on frequency hopping as well as transmission power adjustment as anti-jamming actions. We have implemented and evaluated the effectiveness of our jamming mitigation technique through both physical experiments and NS-3 simulations. The results demonstrate that our proposed technique mitigates jamming by improving packet reception rate, throughput, and energy consumption per packet up to almost 31.83, 30.77, and 34 times, respectively.

8 Acknowledgments

The work was supported by NSF through grants CNS-2301757, CAREER- 2306486, CNS-2306745, and by ONR through grant N00014-23-1-2151.

9 References

[1] http://www.cs.wayne.edu/saifullah/snow.html.
[2] Ettus research. https://www.ettus.com/product/.
[3] GNU Radio. http://gnuradio.org.
[4] Lora alliance. https://lora-alliance.org/.
[5] Ns-3. https://www.nsnam.org/.
[6] Ti CC1310. http://www.ti.com/product/CC1310.
[7] TinyOS. http://www.tinyos.net.
[8] WirelessHART, 2007. https://www.fieldcommgroup.org/technologies/hart.
[9] Fcc first order, 2008. FCC, ET Docket No FCC 08-260, November 2008.
[10] Fcc second order, 2010. FCC, Second Memorandum Opinion and Order, ET Docket No FCC 10-174, September 2010.
[11] K. O. Adefemi Alimi, K. Ouahada, A. M. Abu-Mahfouz, and S. Rimer. A survey on the security of low power wide area networks: Threats, challenges, and potential solutions. *Sensors*, 20(20):5800, 2020.
[12] E. Altman, K. Avrachenkov, and A. Garnaev. Jamming in wireless networks: The case of several jammers. In *Game Theory for Networks, 2009. GameNets' 09. International Conference on*, pages 585–592. IEEE, 2009.
[13] E. Aras, N. Small, G. S. Ramachandran, S. Delbruel, W. Joosen, and D. Hughes. Selective jamming of lorawan using commodity hardware. In *Proceedings of the 14th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, pages 363–372, 2017.
[14] P. Bahl, R. Chandra, T. Moscibroda, R. Murty, and M. Welsh. White space networking with wi-fi like connectivity. In *SIGCOMM '09*.
[15] T. Basar. The gaussian test channel with an intelligent jammer. *IEEE Transactions on Information Theory*, 29(1):152–157, 1983.
[16] R. Chandra, R. Mahajan, T. Moscibroda, R. Raghavendra, and P. Bahl. A case for adapting channel width in wireless networks. In *SIGCOMM '08*.
[17] R. Chen, J.-M. Park, and J. H. Reed. Defense against primary user emulation attacks in cognitive radio networks. *IEEE Journal on selected areas in communications*, 26(1), 2008.
[18] R. Daidone, G. Dini, and M. Tiloca. A solution to the gts-based selective jamming attack on ieee 802.15.4 networks. *Wirel. Netw.*, 20(5):1223–1235, July 2014.
[19] A. Gouissem, K. Abualsaud, E. Yaacoub, T. Khattab, and M. Guizani. Iot anti-jamming strategy using game theory and neural network. In *2020 International Wireless Communications and Mobile Computing (IWCMC)*, pages 770–776. IEEE, 2020.
[20] C. Han, A. Liu, H. Wang, L. Huo, and X. Liang. Dynamic anti-jamming coalition for satellite-enabled army iot: A distributed game approach. *IEEE Internet of Things Journal*, 7(11):10932–10944, 2020.
[21] N. Hou, X. Xia, and Y. Zheng. Jamming of lora phy and countermeasure. In *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*, pages 1–10. IEEE, 2021.
[22] C.-Y. Huang, C.-W. Lin, R.-G. Cheng, S. J. Yang, and S.-T. Sheu. Experimental evaluation of jamming threat in lorawan. In *2019 IEEE 89th Vehicular Technology Conference (VTC2019-Spring)*, pages 1–6. IEEE, 2019.
[23] A. Kashyap, T. Basar, and R. Srikant. Correlated jamming on mimo gaussian fading channels. *IEEE Transactions on Information Theory*, 50(9):2119–2123, 2004.

[24] M. Labib, S. Ha, W. Saad, and J. H. Reed. A colonel blotto game for anti-jamming in the internet of things. In *2015 IEEE global communications conference (GLOBECOM)*, pages 1–6. IEEE, 2015.

[25] L. Lazos, S. Liu, and M. Krunz. Mitigating control-channel jamming attacks in multi-channel ad hoc networks. In *Proceedings of the Second ACM Conference on Wireless Network Security*, WiSec '09, pages 169–180, 2009.

[26] H. Li and Z. Han. Dogfight in spectrum: Combating primary user emulation attacks in cognitive radio systems?part ii: Unknown channel statistics. *IEEE Transactions on Wireless Communications*, 10(1):274–283, 2011.

[27] G. Mao, B. Fidan, and B. D. O. Anderson. Wireless sensor network localization techniques. *Computer networks*, 51(10):2529–2553, 2007.

[28] K. Mikhaylov, R. Fujdiak, A. Pouttu, V. Miroslav, L. Malina, and P. Mlynek. Energy attack in lorawan: Experimental validation. In *Proceedings of the 14th International Conference on Availability, Reliability and Security*, pages 1–6, 2019.

[29] A. Mpitziopoulos, D. Gavalas, C. Konstantopoulos, and G. Pantziou. A survey on jamming attacks and countermeasures in wsns. *IEEE Communications Surveys Tutorials*, 11(4):42–56, 2009.

[30] H. Mustafa, X. Zhang, Z. Liu, W. Xu, and A. Perrig. Jamming-resilient multipath routing. *IEEE Transactions on Dependable and Secure Computing*, 9(6):852–864, 2012.

[31] M. G. Oskoui, P. Khorramshahi, and J. A. Salehi. Using game theory to battle jammer in control channels of cognitive radio ad hoc networks. In *2016 IEEE International Conference on Communications (ICC)*, pages 1–5, 2016.

[32] K. Pelechrinis, C. Koufogiannakis, and S. V. Krishnamurthy. Gaming the jammer: Is frequency hopping effective? In *2009 7th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks*, pages 1–10, 2009.

[33] H. Pirayesh and H. Zeng. Jamming attacks and anti-jamming strategies in wireless networks: A comprehensive survey. *IEEE communications surveys & tutorials*, 24(2):767–809, 2022.

[34] A. Proano and L. Lazos. Packet-hiding methods for preventing selective jamming attacks. *IEEE Transactions on Dependable and Secure Computing*, 9(1):101–114, 2012.

[35] M. Rahman, D. Ismail, V. P. Modekurthy, and A. Saifullah. Implementation of lpwan over white spaces for practical deployment. In *Proceedings of the International Conference on Internet of Things Design and Implementation*, pages 178–189, 2019.

[36] M. Rahman, D. Ismail, V. P. Modekurthy, and A. Saifullah. Lpwan in the tv white spaces: A practical implementation and deployment experiences. 20(4), 2021.

[37] D. R. Raymond and S. F. Midkiff. Denial-of-service in wireless sensor networks: Attacks and defenses. *IEEE Pervasive Computing*, 7(1):74–81, 2008.

[38] A. Saifullah, M. Rahman, D. Ismail, C. Lu, R. Chandra, and J. Liu. Snow: Sensor network over white spaces. In *Proceedings of the 14th ACM Conference on Embedded Network Sensor Systems CD-ROM*, SenSys '16, pages 272–285, New York, NY, USA, 2016. ACM.

[39] A. Saifullah, M. Rahman, D. Ismail, C. Lu, J. Liu, and R. Chandra. Enabling reliable, asynchronous, and bidirectional communication in sensor networks over white spaces. In *Proceedings of the 15th ACM Conference on Embedded Network Sensor Systems*, SenSys '17, pages 9:1–9:14, New York, NY, USA, 2017. ACM.

[40] A. Saifullah, M. Rahman, D. Ismail, C. Lu, J. Liu, and R. Chandra. Low-power wide-area network over white spaces. *IEEE/ACM Transactions on Networking*, 26(4):1893–1906, Aug 2018.

[41] H. B. Salameh, S. Otoum, M. Aloqaily, R. Derbas, I. Al Ridhawi, and Y. Jararweh. Intelligent jamming-aware routing in multi-hop iot-based opportunistic cognitive radio networks. *Ad Hoc Networks*, 98:102035, 2020.

[42] Statista. Number of Internet of Things (IoT) connected devices. https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/: :text=The2021.

[43] R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[44] M. Tiloca, D. D. Guglielmo, G. Dini, G. Anastasi, and S. K. Das. Jammy: A distributed and dynamic solution to selective jamming attack in tdma wsns. *IEEE Transactions on Dependable and Secure Computing*, 14(4):392 –405, August 2017.

[45] C. J. Watkins and P. Dayan. Q-learning. *Machine learning*, 8:279–292, 1992.

[46] A. D. Wood, J. A. Stankovic, and G. Zhou. Deejam: Defeating energy-efficient jamming in ieee 802.15. 4-based wireless networks. In *2007 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, pages 60–69. IEEE, 2007.

[47] Y. Wu, B. Wang, K. J. R. Liu, and T. C. Clancy. Anti-jamming games in multi-channel cognitive radio networks. *IEEE Journal on Selected Areas in Communications*, 30(1):4–15, 2012.

[48] D. Yang, G. Xue, J. Zhang, A. Richa, and X. Fang. Coping with a smart jammer in wireless networks: A stackelberg game approach. *IEEE Transactions on Wireless Communications*, 12(8):4038–4047, 2013.

[49] D. Yang, G. Xue, J. Zhang, A. Richa, and X. Fang. Coping with a smart jammer in wireless networks: A stackelberg game approach. *IEEE Transactions on Wireless Communications*, 12(8):4038–4047, 2013.

[50] Q. Zhu, H. Li, Z. Han, and T. Basar. A stochastic game model for jamming in multi-channel cognitive radio systems. In *2010 IEEE International Conference on Communications*, pages 1–6, 2010.