Rolling Preambles: Mitigating Stealthy FO Estimation Attacks in OFDM-based 802.11 Systems

Hanif Rahbari* and Marwan Krunz†
Department of Electrical & Computer Engineering, The University of Arizona, Tucson, AZ
Email: *rahbari@email.arizona.edu, †krunz@email.arizona.edu

***Abstract*—Modern wireless systems and standards increasingly rely on OFDM for high-throughput communications. However, these systems are often highly vulnerable to selective jamming attacks, particularly when a jammer targets (part of) the known frame preamble. In this paper, we consider one of the most disruptive jamming attacks against the preamble-based frequency offset (FO) estimation in IEEE 802.11a/n/ac/ax systems and develop four techniques to mitigate this attack. Two of these techniques are based on randomly changing the first half of the standard frame preamble at the transmitter while maintaining its *backward compatibility* with legacy receivers. Specifically, we design a set of new preamble waveforms that satisfy the expected characteristics of a preamble in 802.11 systems. The other two techniques take a receiver-based approach and exploit the parts of the preamble that are not under attack to estimate the FO. We conduct extensive simulations and illustrative USRP experiments to study the effectiveness of these countermeasures.**

***Index Terms*—PHY-layer, preamble, frequency offset, OFDM, IEEE 802.11, reactive jamming, mitigation techniques.**

I. Introduction

Wireless systems need to adopt effective preventive measures against possible malicious activities. Jamming is a common denial-of-service (DoS) attack in wireless networks, where an adversary (Eve) injects a jamming signal into the communication medium during a legitimate transmission from a transmitter (Alice) to a receiver (Bob). Jamming attacks can be persistent, random, or selective. In persistent (barrage) jamming attacks, Eve continuously jams the medium, which consumes high jamming power. In contrast, in selective jamming attacks Eve adopts a more energy-efficient strategy by intelligently jamming only valuable packets (e.g., control packets) or highly vulnerable parts of a packet to significantly disrupt the ongoing transmission. Because the jamming duration in selective jamming attacks is short, it is harder for the system to locate and physically avoid or neutralize the jammer. Hence, advanced jamming devices tend to be selective rather than persistent or random.

Modern wireless systems and standards (e.g., LTE, 802.11a/n/ac/ax/ah) increasingly rely on Orthogonal Frequency Division Multiplexing (OFDM). Several selective jamming attacks against OFDM-based systems have been developed in the literature [1]–[8]. In this paper, we explore mitigating such selective jamming attacks. These attacks select and jam different components of an OFDM-based frame, such as the frame preamble [2], [7], [8] or pilot subcarriers [1], and cause different levels of damage to the transmission. One type of selective jamming attacks that causes significant damage to OFDM systems is frequency offset (FO) estimation attack [2], [8]. In particular, the FO estimation attack proposed in [8] has the least jamming duration compared to the durations of other selective attacks, making it extremely difficult to detect and locate the attacker, and does not require knowledge of the channel parameters. In this attack, Eve crafts a fake partial preamble as her jamming signal based on the transmitted frame preamble of an IEEE 802.11a/n/ac/ax system[1]. This preamble is *publicly known*. Using her knowledge of the frame preamble, Eve jams a specific but small portion (less than 17.5%) of the transmitted preamble from Alice to Bob so as to shift the subcarriers by an integer number of the frequency spacing, highly disturbing both the frequency and the channel estimation processes at Bob. If the existence of the attack and the amount of subcarrier shift caused by it are not detected, the attack inflicts almost 50% bit-error rate (BER) to the received payload at Bob. This level of BER is high enough to prevent practical coding schemes from recovering the frame. It is achieved even if the jamming power at Bob is less than the received signal power from Alice. Other FO attack schemes use a random signal for jamming, which necessitates using much higher jamming power and/or duration compared to the above attack.

In general, jamming mitigation is often achieved via spread spectrum techniques. However, these techniques are only applicable on single-carrier systems. To mitigate jamming attacks that target a specific portion of the frame preamble in OFDM systems, the authors in [2] proposed randomizing the preamble's location within a frame. However, inserting the preamble in the middle of a frame delays the decoding of the frame duration field in the physical (PHY) layer header, preventing correct operation at the receiver (Rx). It also increases the error in detecting the start of the preamble because the preamble is prefixed by a signal rather than (often zero-mean Gaussian) noise. The authors in [2] also proposed using the cross-ambiguity function [9] for estimating the amount of subcarrier shift based on only one training symbol. However, this method requires a training symbol whose frequency-domain elements are i.i.d random variables with zero mean. The preamble in IEEE 802.11 systems does not satisfy this requirement.

Contributions– The highly disruptive and channel-independent FO attack in [8] targets parts of the first half of the

[1]The preamble phase warping attack in [2] is a weaker version of this attack, where the jamming signal is a random frequency-shifted version of an arbitrary signal and lasts more than the jamming signal in [8].

preamble to inflict a subcarrier shift. In this work, we first propose a mitigation technique that uses the second half of the publicly known preamble to estimate the amount of a subcarrier shift. Despite its appeal, we show that this mitigation technique has its limitations. Specifically, we explain how Eve can thwart this preliminary countermeasure by extending the fake-preamble jamming signal to the second half of the preamble. This underlines the vulnerability of *any* mitigation technique that relies on a publicly known preamble. Motivated by this fact, we then reconsider the first part of the preamble and propose three preamble randomization techniques, which can be applied independently or jointly at different stages of the communication. Two of these techniques are implemented at the transmitter (Tx), while the last one is implemented at the Rx side. The idea is to make the first part of the preamble somewhat unpredictable via randomization so that Eve can take less benefit of the public knowledge about the standardized preamble. This is possible because in OFDM-based 802.11 systems, the Rx does not necessarily need to know the exact value of the first half of the preamble.

In designing the Tx-side randomization techniques, however, we recognize three constraints that should be satisfied by the new preamble signals. First, because of the widespread use of 802.11 systems and for interoperability purposes, it is important that any Tx and/or Rx that implements these mitigation techniques remains backward-compatible and able to communicate with legacy 802.11 devices. For example, if the Tx uses a non-standardized preamble signal, a legacy 802.11-compliant Rx must still be able to perform regular preamble functions. Second, standardized preambles are designed to satisfy certain properties, including high FO estimation range, good frame detection accuracy, low dynamic range, and low *peak-to-average-power ratio* (PAPR) [10]. Not using the default preamble signal may come at the cost of higher dynamic range and PAPR. Hence, the proposed preamble signals should minimize this cost. Third, the second half of the standard preamble in OFDM-based 802.11 systems is used for channel estimation, and so should be known to Bob. Therefore, that part should not be modified.

The remainder of this paper is organized as follows. In Section II, we provide an overview of the preamble in OFDM-based 802.11 systems and its primary functions, especially FO estimation. We then present in Section III a short description of a stronger variant of the FO attack than the one proposed in [8] and explain how the jamming signal is generated. Our first Rx-based mitigation technique is proposed in Section IV, which is followed by an extension of the FO attack that thwarts this technique. Our proposed set of randomization techniques are then developed in Section V. In Section VI, the results of our simulations and USRP-based experiments are presented. Finally, Section VII concludes the paper.

II. Frame Preamble in OFDM-based 802.11 Systems

Every PHY-layer frame starts with a preamble. In this section, we explain the special characteristics and main operations based on which the standard preamble of an OFDM-based IEEE 802.11 frame is designed.

A. Characteristics of the Preamble in 802.11 Systems

In OFDM, a bitstream is modulated and transmitted over a set of orthogonal frequency channels (subcarriers). In IEEE 802.11 standards, these subcarriers are spaced by $f_\Delta = 312.5$ kHz within the given bandwidth, i.e., 20 to 80 MHz. The preamble in these systems begins with two essential fields, *short training field* (STF) and *long training field* (LTF). Current IEEE standards (e.g., [11]) consider certain characteristics for each of these fields to satisfy the requirements related to various preamble functions. Any modification in this design should take these requirements into account.

The STF contains ten identical short training sequences (STSs), which represent ten replicas of a particular periodic signal with period $\lambda_{STF} = 0.8\,\mu s$, PAPR $R_{\text{PAP}} = 2.24$ dB (in 802.11a/g [10]), and dynamic range $R_{\text{DR}} = 7.01$ dB. Due to the nonlinearity of the power amplifier at the Tx, the PAPR of the STF is design to be as small as possible to avoid poor transmission. Similarly, $R_{\text{DR}} = 7.01$ dB is one of the lowest possible dynamic values among the signals that have low R_{PAP}.

The LTF consists of two long training sequences (LTSs), which represent two cycles of another known periodic signal with period $\lambda_{LTS} = 4\lambda_{STF}$, plus a $1.6\,\mu s$ cyclic prefix. (In MIMO OFDM-based 802.11 systems, these two fields are followed by additional known training sequences for MIMO channel estimation [12]). The minimum subcarrier spacing in the LTF is f_Δ. In contrast, the periodic signal in the STF is constructed by superposing only the subcarriers whose frequencies are integer multiples of $4f_\Delta$. As a result, the minimum subcarrier spacing between any two STS-enabled subcarriers is $4f_\Delta$, and hence their period is $\lambda_{STF} = \lambda_{LTS}/4$.

B. Preamble Operations

The preamble is used for various purposes, including frame detection, time synchronization, FO estimation, automatic gain control (AGC), diversity selection, and channel estimation. In the following, we briefly explain how those functions rely on the special characteristics of the preamble and why these functions perform better under a publicly known preamble.

1) FO and Channel Estimation: The STF is used for frame detection and coarse FO correction. In OFDM, FO can create significant BER at the Rx [13]. The Rx usually uses the preamble to estimate and correct the FO (which is typically the same for all OFDM subcarriers) and adjust the subcarriers to their expected orthogonal frequency bins. The LTF, on the other hand, is used for channel estimation and fine-tuning the coarse STF-based FO estimation. The channel estimation is affected by the FO estimation and so should be performed after coarse FO estimation. The error in channel estimation can grow quadratically as a function of the FO estimation error [14].

Let Δf be the actual frequency offset between a Tx and an Rx. This FO translates into a phase offset of $\Delta\varphi(t) = 2\pi\Delta f\, t$ for the received signal, where t is the time elapsed since the

start of the transmission. The *de facto* time-domain FO estimation method used in OFDM systems is the one developed by Schmidl and Cox [15]. We consider it as a representative but not restrictive FO estimation scheme. It assumes that the channel does not change during the preamble transmission and the preamble (e.g., STF or LTF) is a sequence with two identical halves. Let $\mathbf{r}$ represent such a sequence. The method works as follows. Assume that each half of the sequence has L samples with sampling period of t_s. Let r_i be the ith sample of the sequence $\mathbf{r}$, $i = 1, \ldots, 2L$. So $r_i = r_{L+i}$. Ignoring the noise, this equality also holds for the corresponding samples at the Rx as long as $\Delta f = 0$. However, when $\Delta f \neq 0$, the phase of r_{L+i} relative to r_i is rotated by $\Delta\varphi(t_s) = 2\pi\Delta f L t_s$. Multiplying the conjugate of r_i (i.e., r_i^*) by r_{L+i}, we obtain:

$$s_i \triangleq r_i^* r_{L+i} = |r_i|^2 e^{-j2\pi\Delta f L t_s} = |r_i|^2 e^{-j\Delta\varphi(t_s)}. \quad (1)$$

Taking into account the channel coefficient $h_i = h_{L+i}$ and the noise terms, n_i and n_{L+i}, the value of s_i at the Rx, denoted by $\widetilde{s_i}$, is:

$$\widetilde{s_i} = |h_i r_i|^2 e^{-j2\pi\Delta f L t_s} + \bar{n_i} \quad (2)$$

where $\bar{n_i} \triangleq r_i^* n_{L+i} + n_i^* r_{L+i} + n_i^* n_{L+i}$ has zero mean. Generalizing this calculation to multi-path channel scenarios is straightforward, but after excluding the first few preamble samples. To average out the $\bar{n_i}$'s, the estimated phase offset, $\widetilde{\Delta\varphi}$, is:

$$\widetilde{\Delta\varphi(t_s)} = \measuredangle\Big(\sum_{i=0}^{L-1} \widetilde{s_i}\Big) \quad (3)$$

where the notation $\measuredangle(x)$ indicates the phase of a complex quantity x. Thus, the estimated FO is:

$$\widetilde{\Delta f} = \frac{\widetilde{\Delta\varphi(t_s)}}{2\pi L t_s}. \quad (4)$$

In the 802.11a/g, two of the last three STSs are chosen to form a sequence with two identical halves for coarse FO estimation.

While measuring the phase of a complex number such as $\widetilde{s_i}$, the Rx observes only a value between $-\pi$ and π. Hence, in this method the Rx cannot distinguish $\Delta\varphi$ from $\Delta\varphi \pm 2k\pi$ in (4), for any integer k. The phase offset of 2π corresponds to $\frac{1}{Lt_s}$ offset, i.e., one f_Δ. In general, the phase is unambiguous and correctable as long as $|\Delta f| < \frac{1}{2Lt_s}$ (half a f_Δ). This also implies that a longer period of a cycle reduces the range of FO that can be corrected unambiguously. Given a fixed sampling interval, a longer period results in higher L. Let th_s and th_l be the maximum $|\Delta f|$ values that the STF and the LTF can correct unambiguously, respectively. Since the number of samples of an LTS is four times the number of samples of an STS, then $th_l = th_s/4 = f_\Delta/2$. IEEE 802.11 standard assumes that the maximum FO is always less than th_s [11].

The above discussion reveals a tradeoff between the accuracy and range of the correctable FO. The goal of the STF is to estimate a large FO value and compensate for it by multiplying the rest of the samples (including those obtained during the LTF) by $e^{-j(-2\pi\widetilde{\Delta f_s}\, i\, t_s)}$, where $\widetilde{\Delta f_s}$ is the estimated FO in the STF phase and i is the sample index. Using the LTF, the Rx then computes $\widetilde{\Delta f_l}$ to fine-tune the coarsely estimated FO. This explains one of the reasons for concatenating a training field with a sufficiently short period to a training field with a longer period in 802.11 systems. Consequently, if $|\Delta f| > th_s$, this FO estimation method fails to fully compensate for it.

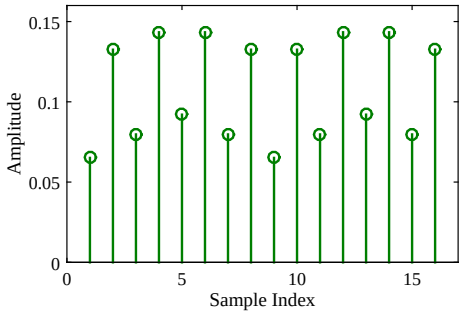


Fig. 1. Amplitude of the samples $r_1, \ldots, r_{16}$ that belong to one STS in the standard STF definition ($t_s = 50\,ns$).

2) Frame Detection: For a typical Rx, an increase in the received power is a first indication of a new frame. To verify whether this increase is indeed due to a transmitted 802.11 frame and then time-synchronize with it, the Rx checks for the existence of a periodic signal with a preset period [15]. In the 802.11 standard, the Rx considers two non-overlapping intervals, each of duration $k\lambda_{STF}$ microseconds (equivalently, kL samples, where $1 \leq k \leq 5$ is an integer) to represent two identical halves of a sequence. The correlation between the samples' conjugates in the first interval (window) and the corresponding samples in the second one is computed. Let $\mathcal{A}(n)$ be the summation of these correlations when the first window starts at the nth sample of the whole sequence:

$$\mathcal{A}(n) = \sum_{i=0}^{L-1} \widetilde{s}_{n+i}^* \widetilde{s}_{n+L+i}. \quad (5)$$

Using $\mathcal{A}(n)$, a normalized timing metric, $\mathcal{M}(n)$, is computed:

$$\mathcal{M}(n) = \frac{|\mathcal{A}(n)|^2}{\big(\mathcal{E}(n)\big)^2} \quad (6)$$

where $\mathcal{E}(n) \triangleq \sum_{i=0}^{L-1} |\widetilde{s}_{n+L+i}|^2$ is the received signal energy over the second window. $\mathcal{M}(n)$ is close to zero if either window does not contain any preamble sample. On the other hand, $\mathcal{M}(n)$ peaks when both windows contain only preamble samples. Ideally, $\mathcal{M}(n)$ should stay constant at the maximum value of 1, as long as both windows are sliding inside the preamble boundaries. So the first time that $\mathcal{M}(n)$ hits the maximum is marked as the beginning of the frame. Because of noise, however, the maximum $\hat{\mathcal{M}} = \max_n \mathcal{M}(n)$ may occur later than the actual preamble start time. To account for this, the algorithm first finds $\hat{\mathcal{M}}$ and then searches for the earliest time before the occurrence of $\hat{\mathcal{M}}$ with an $\mathcal{M}$ value greater than $(1-\epsilon)\hat{\mathcal{M}}$, where $0 < \epsilon < 1$ is a system parameter. That time instant is taken as the beginning of the frame.

One advantage of the default STF signal in 802.11 is that if the Rx computes (5) starting at the true start of the preamble,

the value of $\mathcal{A}(n)$ will be noticeably higher than the value if (5) is computed one sample earlier. This is because the amplitude of the last STS sample is higher than the average amplitude of the STF signal samples (see Fig. 1). Otherwise, if the power of the last sample is less than the average and close to the noise power, that sample will not sufficiently contribute in (5) and the algorithm will likely take the sample before the true start as the beginning of the frame.

3) AGC: The STF is also used for AGC convergence. In order to accelerate AGC locking and adjusting the reference signal value for the A/D converter at the Rx, the dynamic range of the STF should be low so that it can be covered without any overflow/underflow by the A/D converter resolution [10].

III. Overview of Fake-Preamble FO Estimation Attack

Before explaining the proposed mitigation techniques, we first present a stronger variant of the jamming-signal construction in [8], but assume similar to [8] that Eve estimates Eve-to-Bob and Alice-to-Bob FOs, denoted by Δf_{eb} and Δf_{ab}, respectively, via overhearing. She then launches its jamming attack in two phases: (1) Eavesdropping on the channel to pinpoint the start of Alice's frame transmission and acquire its timing information; and (2) jamming the last three STSs of the preamble, which are designated by the standard for coarse FO estimation. We further assume that Eve employs the same *fast frame detection* method as in [8]. During the fast frame detection, Eve estimates the $V = \log_2(L)$ most-probable sample indices $i_0, i_1, \ldots, i_{V-1}$ as the possible frame start times to account for frame detection inaccuracies.

Based on i_0, Eve computes the arrival time of the last three STSs of the preamble and generates a jamming signal that would be aligned with those STSs. The jamming sequence is designed to deceive Bob into erroneously estimate the FO beyond th_l after receiving the STSs, instead of reducing it. This way, not only LTF-based channel estimation will be highly erroneous, but also LTF- and pilot-based FO estimation will fail without needing Eve to jam the LTSs or pilot sub-carriers. For this attack to be successful, Eve further accounts for unknown channel parameters and frame-detection timing errors. The jamming sequence is constructed as follows.

Without loss of generality, Eve assumes i_0 to be the correct start time of the frame (we will relax this assumption later). Let $\Delta\varphi_{ab}$, $\Delta\varphi_{eb}$, and $\Delta\varphi_l = \pi/4$ be the phase offsets corresponding to Δf_{ab}, Δf_{eb}, and th_l, respectively, after a single STS. To cause incorrect FO estimation ($\widetilde{\Delta f_s}$) such that the updated FO after STSs ($\Delta f_{ab} - \widetilde{\Delta f_s}$) is higher than th_l, the following inequality should hold:

$$|\Delta\varphi_{ab} - \widetilde{\Delta\varphi}| > \Delta\varphi_l. \tag{7}$$

Eve's jamming signal needs to satisfy (7). Let g be the Eve-to-Bob channel coefficient. We assume that during Eve's jamming period, g is the same for all the jamming samples that belong to the jamming sequence $\mathbf{u} = u_1, \ldots, u_{2L}$. Let $\tilde{r_i} \triangleq h r_i\, e^{-j2\pi\times i\,\Delta f_{ab} t_s}$ and $\tilde{u_i} \triangleq g u_i\, e^{-j2\pi\times i\,\Delta f_{eb} t_s}$.

To construct the jamming signal, Eve exploits knowledge of the FO estimation algorithm and of Δf_{ab} to construct a fake preamble with "identical halves". For now, assume that the samples of the jamming signal $u_i, i = 1, \ldots, 2L$ can take any arbitrary value. Having identical halves allows Eve to control and carefully calculate a desired FO for $\mathbf{u}$ based on how Bob estimates Δf_{ab}. To explain how Eve calculates it, first consider the superposition of Alice's signal and Eve's jamming at Bob. Dropping the index i from (2) and ignoring the noise term, we have:

$$\tilde{s} = (\tilde{r} + \tilde{u})^*(\tilde{r} e^{-j\Delta\varphi_{ab}} + \tilde{u} e^{-j\Delta\varphi_{eb}}) = e^{-j\Delta\varphi_{ab}} \times \Big[\underbrace{|\tilde{r}|^2 + |\tilde{u}|^2 e^{-j(\Delta\varphi_{eb}-\Delta\varphi_{ab})} + \tilde{r}^*\tilde{u} e^{-j(\Delta\varphi_{eb}-\Delta\varphi_{ab})} + \tilde{u}^*\tilde{r}}_{\mathcal{B}}\Big]. \tag{8}$$

Thus, the estimated phase offset at Bob is:

$$\widetilde{\Delta\varphi} = \measuredangle\tilde{s} = \Delta\varphi_{ab} + \measuredangle\mathcal{B} + \measuredangle\bar{n}. \tag{9}$$

Note that the phase estimation error $\varphi_e \triangleq \measuredangle\mathcal{B}$ is a function of signal-to-jamming ratio (SJR) at Bob and $\Delta\varphi_{eb}$, and jamming will have no effect if $\varphi_e = 0$.

Upon calculating $\widetilde{\Delta\varphi}$ and $\widetilde{\Delta f_s}$, Bob changes the FO for the rest of the frame to $\Delta f_{ab} - \widetilde{\Delta f_s}$. According to (7), Eve is successful if she can ensure that $\Delta\varphi_{eb}$ satisfies the following:

$$|\Delta\varphi_{ab} - \widetilde{\Delta\varphi}| > \Delta\varphi_l \Rightarrow |\varphi_e + \measuredangle\bar{n}| > \Delta\varphi_l = \frac{\pi}{4}. \tag{10}$$

Eve can guarantee a desired φ_e only if SJR$\rightarrow -\infty$.

However, the phase and amplitude of $\tilde{r}$ are channel-dependent and Eve cannot estimate the Alice-to-Bob channel coefficient h. To address this challenge, Eve takes advantage of Alice's known preamble samples and the product sum in (3) to cancel out the terms with unknown phases. Without loss of generality, let Eve pair the samples in order and let (u_1, u_2) be the first pair of samples in the jamming sequence. By knowing the preamble sample values at Alice and taking into account the phase offsets due to Δf_{ab} and Δf_{eb}, u_2 can be designed such that all the terms that depend on $\tilde{r}$ (excluding $|\tilde{r}|$) in the term $\mathcal{B}$ in (8) are eliminated. That means, the jamming sequence $\mathbf{u}$ must satisfy $\tilde{r}_1\,\tilde{u}_1^* + \tilde{r}_2\,\tilde{u}_2^* = 0$. Different from the jamming sequence construction method in [8], in this paper we take into account the small FO-specific phase changes $2\pi\,\Delta f_{ab} t_s$ and $2\pi\,\Delta f_{eb} t_s$ from $\tilde{r}_1$ and $\tilde{u}_1$ to $\tilde{r}_2$ and $\tilde{u}_2$, respectively, and design a more effective jamming signal. Thus, Eve sets u_2 as follows:

$$u_2 = -\frac{r_1^*\, u_1}{(r_2\, e^{-j2\pi(\Delta f_{ab}-\Delta f_{eb})t_s})^*} \tag{11}$$

which implies that

$$\widetilde{s_1} + \widetilde{s_2} = e^{-j\Delta\varphi_{ab}} \times \Big[|\tilde{r}_1|^2 + |\tilde{r}_2|^2 + (|\tilde{u}_1|^2 + |\tilde{u}_2|^2|)\, e^{-j(\Delta\varphi_{eb}-\Delta\varphi_{ab})}\Big]. \tag{12}$$

The requirement in (11) is similarly imposed on the rest of the even samples of $\mathbf{u}$. Accordingly, the autocorrelation

function $\mathcal{A}$ for this scheme, denoted by $\mathcal{A}_{\text{fake}}$, becomes:

$$\mathcal{A}_{\text{fake}} \triangleq \sum_{i=0}^{L-1} \widetilde{s_i} = e^{-j\Delta\varphi_{ab}} \underbrace{\Big[\sum_{i=0}^{L-1} |\tilde{r}_i|^2 + \sum_{i=0}^{L-1} |\tilde{u}_i|^2 \, e^{-j(\Delta\varphi_{eb}-\Delta\varphi_{ab})} \Big]}_{\mathcal{C}}. \quad (13)$$

Note that the term $\mathcal{C}$ is a function of $(\Delta\varphi_{eb} - \Delta\varphi_{ab})$.

Now Eve can determine a desired value of $\Delta\varphi_{eb}$ in a way that makes $|\measuredangle\mathcal{C}| > \Delta\varphi_l$, which satisfies (10). For a given SJR value, the optimal $|\Delta f_{eb} - \Delta f_{ab}|$ that maximizes $|\measuredangle\mathcal{C}|$ during the STSs was derived in [8]. If $\Delta\varphi_{eb}$ is not optimal, Eve can augment the hardware-dependent Δf_{eb} and obtain an *effective* Δf_{eb} by imposing an artificial FO of Δf_n on the jamming sequence before it is transmitted by the oscillator.

To account for frame detection errors, in the fake preamble attack [8], Eve benefits from the remaining unassigned samples (i.e., odd samples) to cancel out channel-dependent terms for the cases where one of the $V-1$ other possible start times is the true one. By exploiting knowledge of Δf_{ab}, Δf_{eb}, and the transmitted samples **r**, Eve assumes that **u** will be aligned with a cyclically shifted version of **r** and accordingly assigns the values of the remaining jamming sequence samples to eliminate the channel-dependent terms. The amount of cyclic shift is specified by $i_v - i_0$, $v = 1, \ldots, V-1$. Note that in this paper, in designing **u** we consider the phase changes due to FOs along the Alice's and Eve's signals, and create a stronger variant of the attack in [8].

A. Effects of LTSs on FO and Channel Estimation

LTSs are used for fine-tuning the estimated FO and for channel estimation. As explained in Section II, the phase offset from the LTF-based FO correction perspective is between $-\pi$ and π, which means that the true FO after STF-based correction has to be between $-th_l$ and th_l. So LTSs can correct up to $th_l = f_\Delta/2$ FO, and any remaining phase offset will be an integer multiple of 2π, which corresponds to $2k\,th_l = kf_\Delta$, $k = 1, 2, \ldots$. In other words, the LTSs at Bob round up the manipulated FO (by $\widetilde{\Delta f_s}$) to the nearest multiple of $2th_l$ by adjusting the subcarriers to the closest, though incorrect, frequency bins. Consequently, in this attack all the subcarriers will be shifted forward or backward, replacing neighboring subcarriers. Bob eventually demodulates the bits of all OFDM symbols, but he is unaware that these symbols have been shifted and misplaced. Therefore, when the bits of different OFDM symbols are concatenated to reconstruct the original bit sequence, the entire sequence will look shuffled and out-of-order compared to the original bit sequence (assuming the LTF-based channel estimation is error-free). A shifted version of an arbitrary bit sequence will result in very high BER.

An STF-based FO estimation error also affects the channel estimation process, which is applied across the LTF. To elaborate, the phase offset accumulates over time, causing different LTS samples to have different phase offsets. However, Bob complacently tries to interpret this time-varying phase offset as a fixed-value channel phasor. Hence, his attempt to model the FO as if it is a channel parameter results in an incorrect estimated channel phasor, which after equalization rotates the payload's modulation symbols on the constellation map. Note that if Eve jams the LTF instead of the STF, she can only degrade the fine FO and channel estimation by using higher jamming power, but still cannot inflict a subcarrier shift. Using the coarsely estimated FO, Bob still can exploit the pilot subcarriers for better FO and channel estimation.

IV. LTF-Based Countermeasure and Its Limitations

In this section, we propose a preliminary countermeasure that exploits the phase differences between the known LTF subcarriers for estimating the amount of subcarrier shift.

When the STF is jammed, Bob has to correct the FO of the LTF before using it to estimate the channel. Otherwise, the channel estimate will be highly erroneous. That implies that the symbols transmitted over LTF subcarriers are still impaired by unknown channel parameters and so Bob cannot find the FO or the amount of subcarrier shift by comparing the received LTF symbols with the transmitted ones. Moreover, because the STF is jammed, existing subcarrier-shift estimation techniques (e.g., [15]) that assume "known" phase changes between the subcarriers of two distinct training symbols would perform poorly. Hence, we explore the characteristics of a single training symbol (i.e., the LTF) that are not impacted by the channel and can be used for estimating the amount of shift.

Differential modulation is a known technique to circumvent channel and FO [16]. In this technique, the phase of a symbol is recursively determined by the current input data and the phase of the previous symbol, and can be applied in both time and frequency domains. The authors in [16] proposed cyclically differential modulation in the frequency domain, i.e., across OFDM subcarriers, to facilitate estimating the integer component of the normalized FO by using only one OFDM symbol. Assuming a coherence bandwidth larger than f_Δ, the sequence of phase differences between every pair of adjacent subcarriers will not be impacted by the channel, and so can be used to estimate the amount of subcarrier shift. Let $\mathcal{S}$ and $\tilde{\mathcal{S}}$ be the sequence of phase differences in one LTF OFDM symbol at Alice and Bob, respectively. In 802.11 systems, $\mathcal{S}$ consists of the phases of BPSK symbols. A cyclically shifted version of $\mathcal{S}$ that has the highest correlation with $\tilde{\mathcal{S}}$ determines the amount of subcarrier shift. To make this technique effective, the authors proposed using Maximal Sequences as $\mathcal{S}$ because in these sequences, the relative magnitude of the off-peak correlation value to the peak correlation value is low [16]. However, the sequence $\mathcal{S}$ in the standard LTF does not completely satisfy the low off-peak correlation property, and so in some scenarios (e.g., low SNR) may perform poorly in estimating the amount of shift.

Let $|\mathcal{S} \odot \tilde{\mathcal{S}}|$ be the cross-correlation value, where $\odot$ is inner-product operator. In Fig. 2, we show the cross-correlation value for different shifted versions of $\tilde{\mathcal{S}}$ under an AWGN channel

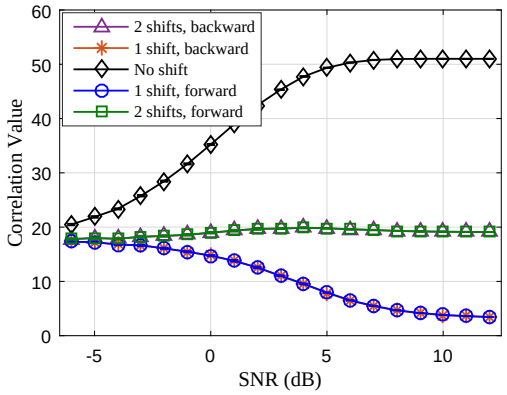


Fig. 2. Cross-correlation of $\mathcal{S}$ and $\tilde{\mathcal{S}}$ for different amounts of subcarrier shift. 95% confidence intervals are too small to be visible.

model. Because the fake-preamble FO estimation attack results in one or two shifts in either forward or backward direction, we only consider these five cases. The simulation results show that this technique is quite effective in estimating the amount of shift, but it cannot determine its direction (forward or backward). Bob will need to try each of the forward and backward shifts and check which one is appropriate to decode the PHY-layer header. In low SNR regime (e.g., SNR < -4 dB), however, the estimation is more prone to errors.

Nevertheless, a more critical limitation of the above countermeasure is that Eve can thwart it by extending her jamming signal to the LTF. In this case, the total jamming duration will be less than $10\,\mu$s and so it is still hard to locate such a short-lived jamming attack. In the following, we show how a jamming signal can be designed to exploit the publicly known sequence $\mathcal{S}$ and corrupt the correlation $|\mathcal{S} \odot \tilde{\mathcal{S}}|$. Recall that one advantage of the differential modulation that is exploited by the above countermeasure is that it is channel-independent (provided that the channel's coherent bandwidth is large enough). Similar to Bob, Eve can take benefit of the same advantage and generate a channel-independent jamming signal whose phase differences corrupt $\tilde{\mathcal{S}}$. Our idea is to add jamming symbols on Alice's LTF subcarriers such that

$$|\mathcal{S} \odot \tilde{\mathcal{S}}^{(j)}| \approx 0 \tag{14}$$

where $\tilde{\mathcal{S}}^{(j)}$ is the sequence of phase differences at Bob when Eve's signal is superposed onto Alice's signal. One straightforward solution to solve (14) is to flip the sign of a particular subset of BPSK symbols in $\tilde{\mathcal{S}}$. We show the feasibility of such an attack through an example, but leave further evaluation of this attack for future work. Consider two LTF subcarrier symbols $sc_1 = 1$ and sc_2, where $sc_2 = sc_1 \times e^{-j\pi} = -1$. The phase difference between them corresponds to the phase of the BPSK symbol -1. Now, suppose that jamming symbols $sc_1^{(j)} = sc_2^{(j)} = 1$ are added to sc_1 and sc_2, respectively. At Bob, this implies receiving $g \times sc_1^{(j)} + h \times sc_1$ and $g \times sc_2^{(j)} + h \times sc_1 \times e^{-j\pi}$ on the two subcarriers, respectively, where g is the unknown Eve-to-Bob channel coefficient. As long as $|g| > |h|$, the phase difference between these two symbols will be less than $\pi/2$ and so the sign of the phase difference in $\tilde{\mathcal{S}}^{(j)}$ is flipped. So the product of this value and its corresponding term in $\mathcal{S}$ will be negative, reducing (14).

V. STF-Based Mitigation Techniques

The enhanced FO estimation attack presented in Section III exploits the facts that: (1) the STF signal is publicly known, and (2) Bob uses the last three STSs for coarse FO estimation. To counter this attack, we propose three randomization techniques that aim at making the STF signal or FO estimation process at Bob unpredictable. However, if the new STF signals do not satisfy the key characteristics that a legacy but legitimate Rx expects, the proposed STF signals are no longer backward-compatible. Furthermore, even if both Alice and Bob are aware of the randomization techniques but Alice's STF signal for each frame is selected randomly to confuse Eve, Bob will also be confused in the absence of a prior per-frame handshaking mechanism, whose implementation is extremely challenging. Our main idea is to exploit the fact that Bob does not need to know the exact STF signal. All he needs to know during the STF for frame detection and FO estimation is the period $\lambda_{STF} = 0.8\,\mu$s of any transmitted periodic STF signal.

A. Shifting the Standard STF in Time

Our first randomization technique is to cyclically shift in time the default STF signal by a random integer-multiple of t_s. As long as the amount of shift does not belong to the set $\{i_v - i_0 | v = 0, \ldots, \log_2(L) - 1\}$, which is assumed by Eve as the possible amount of shifts in $\mathbf{r}$ to account for frame detection errors, the jamming signal cannot effectively eliminate the channel-dependent parameters in (8). Hence, the attack is mitigated. In addition, a time shift does not change the PAPR, the dynamic range, and the period of the default STF signal. Hence, it does not need extra power amplifier capabilities at existing transmitters and is backward compatible with legacy receivers.

However, a time shift may come at the cost of reduced frame detection accuracy because the amplitude of the last STS sample in the shifted sequence in some cases is less than the one of the last sample in the default sequence (Fig. 1). In a noisy channel, this results in one or two sample offset in frame detection. To account for this additional error, Bob in our scheme relies on the LTF-based channel estimation, where the LTF signal is known to Bob, to estimate the amount of the sample offset in the LTF and compensate for it[2]. Note that the LTF signal is not modified in our scheme. We also note that although multiplying the default STF signal by a constant complex number with unit amplitude but random phase will preserve the PAPR, the dynamic range, and the period of the default STF signal, in the case of frame preamble attack Eve can still use the same jamming sequence $\mathbf{u}$ and achieve

[2]Once the STF has been detected, Bob constructs a *Toeplitz* matrix whose first row is filled by the known LTF and the remaining rows are filled with shifted versions of the LTF, each corresponding to one of the possible frame detection errors, i.e., sample offsets. Using the Toeplitz matrix, Bob then estimates the channel coefficient corresponding to each row. The row with the minimum channel estimation error determines the amount of sample offset.

index	-24	-20	-16	-12	-8	-4	4	8	12	16	20	24	R_{PAP}	R_{DR}
$\mathcal{F}(\mathbf{r})$	$1+j$	$-1-j$	$1+j$	$-1-j$	$-1-j$	$1+j$	$-1-j$	$-1-j$	$1+j$	$1+j$	$1+j$	$1+j$	2.24 dB	7.01 dB
$\mathcal{F}(\mathbf{r}^{(1)})$	$1+j$	$1+j$	$1-j$	$1+j$	$1+j$	$1-j$	$-1+j$	$-1-j$	$1+1j$	$1-j$	$1+j$	$-1-j$	2.92 dB	6.51 dB
$\mathcal{F}(\mathbf{r}^{(2)})$	$1+j$	$1+j$	$-1+j$	$1+j$	$1+j$	$-1+j$	$1-j$	$-1-j$	$1+j$	$-1+j$	$1+j$	$-1-j$	2.92 dB	6.51 dB
$\mathcal{F}(\mathbf{r}^{(3)})$	$1-j$	$-1-j$	$-1-j$	$1+j$	$1-j$	$1+j$	$1-j$	$-1-j$	$1-j$	$1-j$	$-1+j$	$-1-j$	2.94 dB	10.03 dB
$\mathcal{F}(\mathbf{r}^{(4)})$	$-1+j$	$1+j$	$-1-j$	$-1-j$	$-1+j$	$-1-j$	$1-j$	$-1-j$	$1-j$	$-1+j$	$-1+j$	$-1-j$	2.94 dB	10.03 dB

TABLE I

Symbols transmitted over the 12 subcarriers of the STF. The notation $\mathcal{F}(\mathbf{x})$ denotes the frequency-domain representation of a time-domain sequence $\mathbf{x}$ after applying FFT. The indices are defined according to the 802.11a notation [11].

the same success. The reason is that the division in (11) cancels out this constant coefficient. Therefore, multiplying the transmitted STF by a constant does not mitigate the attack.

B. Constructing New STFs with Low PAPR/Dynamic Range

The standard preamble was ratified in 1999 and required very low PAPR and dynamic range. However, after about two decades many modern wireless devices are capable of processing signals with higher R_{PAP} and R_{DR} values. For example, COTS wireless routers are usually able to support R_{DR} as large as $100\,\text{dB}$. Motivated by this fact, our second mitigation technique is to use the signals with the same period as the default STF signal, but *slightly* higher R_{PAP} (or R_{DR}). For example, assuming that the Tx can support $R_{\text{PAP}} = 2.92\,\text{dB}$, we can identify two new periodic signals with $R_{\text{DR}} = 6.51\,\text{dB}$. Additionally, by allowing R_{DR} to increase to $10.03\,\text{dB}$, we were able to identify two more periodic signals with $R_{\text{PAP}} = 2.94\,\text{dB}$ (see Table I). Let these four signals be represented by $\mathbf{r}^{(1)}$, $\mathbf{r}^{(2)}$, $\mathbf{r}^{(3)}$, and $\mathbf{r}^{(4)}$. Our second randomization technique is to let Alice randomly select for each frame one of $\mathbf{r}^{(1)}$, $\mathbf{r}^{(2)}$, $\mathbf{r}^{(3)}$, $\mathbf{r}^{(4)}$, and $\mathbf{r}$ as the STF signal. When one of the new signals is selected, the unknown (channel-dependent) parameters in (8) are not fully eliminated because the jamming sequence $\mathbf{u}$ is not designed based on the underlying STF, and so the attack success rate decreases.

As discussed before, any cyclically time-shifted version of a signal will have the same PAPR, dynamic range, and period as the original signal. In contrast to the first mitigation technique, where the random time shift results in the degradation of frame detection accuracy, in the second technique one can maintain (or improve) the frame detection accuracy by using those time-shifted versions of $\mathbf{r}^{(1)}$, $\mathbf{r}^{(2)}$, $\mathbf{r}^{(3)}$, and $\mathbf{r}^{(4)}$ whose first and last samples have low and high amplitude, respectively. Nevertheless, if Bob is able to account for the reduced frame detection accuracy (e.g., by using LTF-based channel estimation), Alice can also arbitrarily shift in time these new STFs to further randomize the preamble. In this case, Eve's jamming signal will be successful in eliminating the channel-dependent parameters only in $\frac{\log_2(L)}{5L}$ of the STF signals.

C. Sequence Hopping

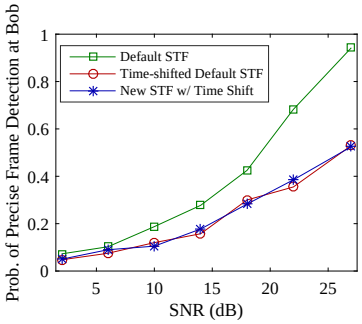
The above two randomization techniques are applicable at the Tx side. At the Rx side, Bob can exploit the redundancy in the STSs and randomly choose any pair of consecutive STSs, including the ones in the last three STSs, to perform FO estimation. Furthermore, due to the maximum FO requirement for 802.11-compliant devices (212 kHz $= 1.3568\,th_s$ for devices operating in the 5 GHz band and 125 kHz $= 0.8\,th_s$ for devices in the 2.4 GHz band [12]), the two autocorrelation windows do not necessarily need to be contiguous. In fact, as initially proposed in [8], the two windows can be two or four STSs apart (i.e., each sample is three or five STSs away from its dual) in the 5 GHz and 2.4 GHz bands, respectively. This means that if Eve jams only three STSs, Bob has the flexibility to randomly hop to any pair of STSs for FO estimation, given that the STSs in this pair are not more than two or four STSs apart, depending on the frequency band. Even if Bob selects an STS that is corrupted by a jamming signal together with a jamming-free one, he is still able to estimate the same FO as if two jamming-free sequences are selected [17]. However, in a multi-path channel environment, the first few received STSs at Bob (up to the time instance that is less than the delay spread of the channel) are not the same as the remaining STSs and so should be excluded from the FO estimation process. In OFDM-based 802.11 systems, it is assumed that the maximum delay spread is less than $1.6\,\mu$s, which is equivalent to two STSs.

To implement sequence hopping, Bob can record the received signal while he is in the process of detecting the start of the frame. Once the frame has been detected and the ten STSs recorded, Bob randomly chooses two STSs for FO estimation, while satisfying the maximum STS-distance constraint.

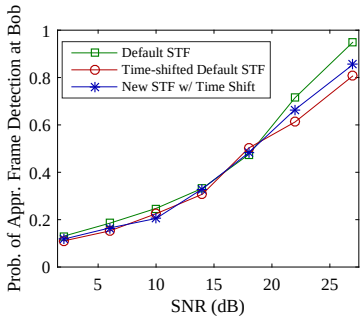
Having said that, one may argue that Eve can expand the jamming duration to cover more STSs and overcome the proposed sequence hopping technique. In this case, we note that Eve has to receive at least two STSs ($1.6\,\mu$s long) and then wait an additional time for switching from receiving to transmitting before it starts jamming. In fact, to the best of our knowledge, the minimum reaction time of a correlation-based reactive jammer demonstrated on the USRP's FPGA is $2.56\,\mu$s [18]. So by the time Eve starts jamming, at least first five STSs have already arrived at Bob. If Eve further employs a received power-based frame detection using only a few samples of the first STS and can jam the whole STF, then either the Tx-based randomization techniques or the STSs-bypassing technique proposed in [8] can be employed.

VI. Performance Evaluation

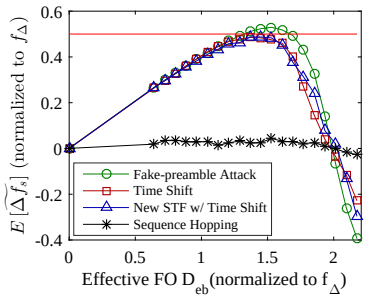
Now we evaluate the effectiveness of the proposed STF-based countermeasures in mitigating the enhanced FO estimation attack through LabVIEW simulations and USRP experiments. Specifically, we measure $\widetilde{\Delta f_s}$ as well as the final estimated FO after the LTF. Frame detection accuracy is also evaluated to study the impact of the Tx-based countermeasures on the legacy systems. Bob uses the first six STSs (out of ten) for frame detection while Eve uses only the first two. We further let Bob use the last two STSs for coarse FO estimation.



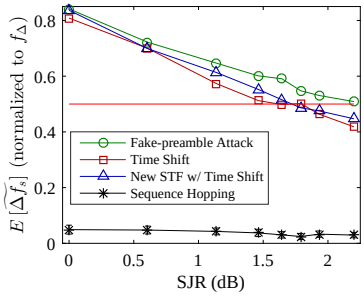
(a) Probability of "precise" frame detection at Bob under different Tx-based countermeasures.



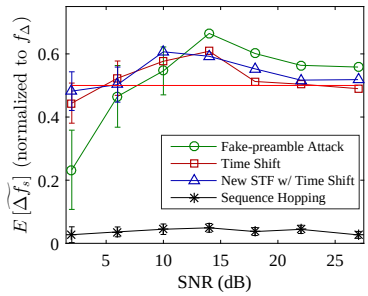
(b) Probability of approximately detecting the frame (with up to two samples error) at Bob under different Tx-based countermeasures.



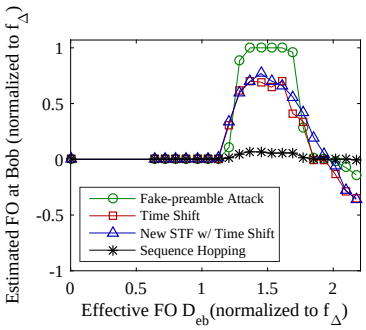
(c) Impact of the D_{eb} on the coarse FO estimation at Bob (SJR= 1.64 dB and SNR= 27 dB).



(d) Impact of the SJR on the coarse FO estimation at Bob (SNR= 18 dB and $D_{eb} = 1.53\, f_\Delta$).



(e) Impact of the SNR on the coarse FO estimation at Bob (SJR= 1.46 dB and $D_{eb} = 1.52\, f_\Delta$).



(f) Impact of the D_{eb} on the overall estimated FO at Bob (SJR= 1.46 dB and SNR= 27 dB).

Fig. 3. Performance of the proposed STF-based mitigation techniques under different noise levels, D_{eb}, and SJR values (simulation results).

We first evaluate the performance under a simulated channel model and later in a multi-path indoor environment.

A. Simulations

We simulate an AWGN channel without signal attenuation and vary the SJR, the SNR, and Eve's effective FO, denoted by D_{eb}. First, we study the impact of our proposed Tx-based countermeasures on the frame detection performance at Bob or any legacy Rx (e.g., Eve). In Fig. 3(a), we plot the probability that Bob precisely detects the frame. (Note that an Rx can use more STSs to improve the accuracy.) The figure shows that shifting the STF signal in time noticeably reduces the accuracy, especially at high SNR values. Likewise, Eve will experience higher rate of frame synchronization errors. However, while Bob can account for this reduction by using LTF-based channel estimation, Eve starts jamming before the LTF and so cannot take benefit of the LTF for more accurate frame detection. Next, we show in Fig. 3(b) the successful frame detection probability when the LTF-based channel estimation can correct up to two sample errors. To compute the detection probability in this figure, we include the cases where the detection is precise as well as those in which Bob detects the frame one or two samples earlier. The results imply that the proposed mitigation techniques often incur only one or two sample errors, which can be accounted for by using the LTF.

Next, we set $\Delta f_{ab} = 0$ and vary D_{eb} under different schemes, and measure $\widetilde{\Delta f_s}$ (using the corrupted STSs) and the final FO estimated after the LTF. Fig. 3(c)-(e) depict the average $\widetilde{\Delta f_s}$, where the horizontal line represents the threshold th_l/f_Δ. While the enhanced fake-preamble attack can satisfy (7) and pass the threshold even at high SJR= 2 dB (see Fig. 3(c) and Fig. 3(d)), the Tx-based countermeasures are able to slightly reduce the FO estimation error and mitigate the attack. Because the PAPR and dynamic range of the new STFs, i.e., $\mathbf{r}^{(1)}$, $\mathbf{r}^{(2)}$, $\mathbf{r}^{(3)}$, and $\mathbf{r}^{(4)}$, are required to be close to the ones of $\mathbf{r}$, these STFs are often similar to $\mathbf{r}$ and so replacing $\mathbf{r}$ with one of them does not significantly mitigate the attack. We suspect that further relaxation of the PAPR and dynamic range constraints would improve the Tx-based countermeasures. On the other hand, sequence hopping technique significantly thwarts the attack and can be considered as a viable solution when the Tx cannot afford high PAPRs and dynamic ranges. We also evaluate the effect of the noise (SNR) in Fig. 3(e). While high noise level (e.g., SNR$<$ 10 dB) can be beneficial for Bob in avoiding the attack, proposed techniques can mitigate the attack at higher SNRs.

In Fig. 3(f), we set SJR= 1.46 dB and plot the final FO computed after the jamming-free LTF. When $|\widetilde{\Delta\varphi}| > \Delta\varphi_l$, the LTSs round the estimated FO to the nearest multiple of $2th_l$. So the curve that belongs to the enhanced FO attack, whose success mainly depends on D_{eb}, alternates between 0, 1, and -1. However, when the Tx-based randomization techniques are employed, we observe about 23% reduction in average FO estimate even at optimal D_{eb}. This implies that they can effectively alleviate the attack in a subset of cases where the randomly-selected STF signal is very different from the default STF signals. Again, the Rx-based randomization technique proves to be very effective, assuming that only the last three STSs are being jammed.

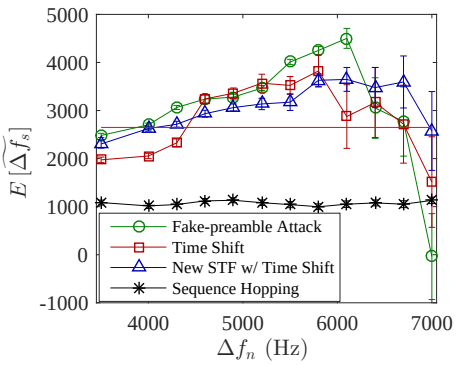


Fig. 4. Effect of Δf_n on STF-based FO estimation under different countermeasures (USRP results).

B. USRP Experiments

We demonstrate the impact of the proposed mitigation techniques using an NI-USRP 2922 testbed, operated in an indoor environment. Our setup consists of three USRPs, acting as Alice, Bob, and Eve. In [8], we explain our techniques to overcome the challenges of implementing our reactive jamming attack on the USRP. In our experiments, $f_\Delta = 3125$ Hz, and Δf_{ab} and Δf_{eb} were measured to be 1086 and 340 Hz with standard deviations 270 and 230 Hz, respectively. Alice-Bob, Alice-Eve, and Eve-Bob distances are $2.1\,m$, $1.88\,m$, and $1.68\,m$, respectively. We set Alice's and Eve's transmission powers to 7.85 dBm and 11 dBm, respectively, and vary Eve's effective FO by varying Δf_n.

Fig. 4 shows the average STF-based estimate of Δf_{ab} ($\widetilde{\Delta f_s}$) for different values of Δf_n. The attack is successful if $\widetilde{\Delta f_s} > \Delta f_{ab} + f_\Delta/2$. The horizontal line represents the most probable value of $\Delta f_{ab} + f_\Delta/2$. The sequence hopping mitigation technique outperforms other countermeasures and effectively neutralizes the attack. Large confidence intervals imply that the Tx-based countermeasures (i.e., time shift and new STF signals) mitigate the attack in some cases, but not to the extent that totaly neutralize it.

VII. Conclusion

In this work, we explored four countermeasures to mitigate one of the most devastating jamming attacks against OFDM-based 802.11 systems. This and other similar attacks target the frame preamble of these systems and disrupt the frequency offset estimation. We proposed two techniques to randomize the first half of the standard preamble by constructing new preamble waveforms in a way that the expected characteristics of the preamble are almost preserved. Such a design allows the new transmitters to maintain their backward compatibility with legacy receivers. We also proposed two receiver-based mitigation techniques that exploit the jamming-free components of the known preamble to mitigate the attack and discussed their limitations, especially when a countermeasure relies on a publicly known preamble. Our simulations and USRP experiments show that receiver-based approaches perform better than the transmitter-based approaches due to the tight characteristics of the preamble. Future work can involve studying the capabilities of new wireless devices and accordingly developing new preamble characteristics that can provide more flexibility in designing rolling preambles.

Acknowledgement

This research was supported by NSF (grants CNS-1409172, IIP-1432880, and IIP-1535573) and ARO (grant W911NF-13-1-0302). Any opinions, findings, conclusions, or recommendations expressed in this paper are those of the author(s) and do not necessarily reflect the views of NSF or ARO.

References

[1] T. C. Clancy, "Efficient OFDM denial: Pilot jamming and pilot nulling," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Kyoto, Japan, Jun. 2011.

[2] M. J. LaPan, T. C. Clancy, and R. W. McGwier, "Phase warping and differential scrambling attacks against OFDM frequency synchronization," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Vancouver, BC, Canada, May 2013, pp. 2886–2890.

[3] C. Shahriar, S. Sodagari, R. McGwier, and T. Clancy, "Performance impact of asynchronous off-tone jamming attacks against OFDM," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Budapest, Hungary, Jun. 2013, pp. 2177–2182.

[4] C. Mueller-Smith and W. Trappe, "Efficient OFDM denial in the absence of channel information," in *Proc. IEEE Military Commun. Conf. (MILCOM)*, San Diego, CA, USA, Nov. 2013, pp. 89–94.

[5] C. Shahriar *et al.*, "PHY-Layer resiliency in OFDM communications: A tutorial," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 1, pp. 292–314, 2015.

[6] H. Li and X. Wang, "Disguised jamming against OFDM transmission through nonlinear amplify-and-forward," in *Proc. IEEE Military Commun. Conf. (MILCOM)*, Tampa, FL, USA, Oct. 2015, pp. 576–580.

[7] M. J. LaPan, T. C. Clancy, and R. W. McGwier, "Physical layer orthogonal frequency-division multiplexing acquisition and timing synchronization security," *Wireless Commun. Mobile Comput.*, vol. 16, no. 2, pp. 177–191, Feb. 2016.

[8] H. Rahbari, M. Krunz, and L. Lazos, "Swift jamming attack on frequency offset estimation: The Achilles' Heel of OFDM systems," *IEEE Trans. Mobile Comput.*, vol. 15, no. 5, pp. 1264–1278, May 2016.

[9] D. Li, Y. Li, H. Zhang, and L. Wang, "Cross ambiguity function based integer frequency offset estimation for OFDM systems," in *IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2012, pp. 1065–1069.

[10] R. Boehnke and T. Doelle, "Alternative proposal for BRAN SYNCH preamble," Mar. 1999, doc.: IEEE 802.11-99/048. [Online]. Available: http://goo.gl/NIy6BM

[11] "IEEE Std 802.11a-1999," *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: High-Speed Physical Layer in the 5 GHz Band*, 1999.

[12] "IEEE Std 802.11n-2009," *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 5: Enhancements for Higher Throughput*, 2009.

[13] T. Pollet, M. Van Bladel, and M. Moeneclaey, "BER sensitivity of OFDM systems to carrier frequency offset and Wiener phase noise," *IEEE Trans. Commun.*, vol. 43, no. 234, pp. 191–193, 1995.

[14] L. Weng, R. Murch, and V. Lau, "SISO-OFDM channel estimation in the presence of carrier frequency offset," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, vol. 3, Las Vegas, NV, USA, Apr. 2006, pp. 1444–1449.

[15] T. M. Schmidl and D. C. Cox, "Robust Frequency and Timing Synchronization for OFDM," *IEEE Trans. Commun.*, vol. 45, no. 12, pp. 1613–1621, Dec. 1997.

[16] Z. Liu, B. Weng, and Q. Zhu, "Frequency offset estimation for differential OFDM," *IEEE Trans. Wireless Commun.*, vol. 4, no. 4, pp. 1737–1748, 2005.

[17] H. Rahbari, M. Krunz, and L. Lazos, "Security vulnerability and countermeasures of frequency offset correction in 802.11a systems," in *Proc. IEEE INFOCOM*, Toronto, ON, Canada, Apr. 2014, pp. 1015–1023.

[18] D. Nguyen *et al.*, "A real-time and protocol-aware reactive jamming framework built on software-defined radios," in *Proc. ACM Workshop Softw. Radio Implementation Forum (SRIF)*, Chicago, IL, USA, Aug. 2014, pp. 15–22.