ISSN: 2320-1363

DETECTION OF JAMMING IN QPSK BASED FHSS WITH BER METRIC

L.Nandhini[*1], N.C. Sendhil kumar[*2]

M.E Scholar, Communication Systems, Dept. of ECE, Ranippettai Engineering College, Walajah, India

Asst. Professor & Head, Dept. of ECE, Ranippettai Engineering College, Walajah, India

ABSTRACT

Though active jammers are alive at all times while switching ON, reactive jammers are considered to be a serious threat for wireless communication. Along with this, it is difficult to detect their presence reliably. We propose a novel method to detect such sophisticated jammers in frequency hopping spread spectrum (FHSS) wireless communication systems. The key idea is to extract statistics from the jamming-free symbols of the FHSS synchronizer to discern jammed packets from those lost due to bad channel conditions along with phase corrections. Our contribution is based on two steps. First, we detect the presence of jamming by evaluating preamble symbols of IEEE 802.15.4 packets, thus enabling the accurate prediction of the packet delivery ratio (PDR), after the hop sequence is detected correctly. We introduce another metric called spectral efficiency, to detect the frequency hops correctly, even at jammed condition and hence correct retrieval of packets. Our second work is the design and evaluation of a detection technique relying on this metric to detect reactive jammers. We build a software-defined radio test bed and show that our technique enables the error-free detection of reactive jammers that jam all packets on links with a PDR above 0.3. Implementation under subcarrier hops sequence. This is considered to be a valid work under jamming related issues. The simulations are performed using MATLAB software with 2013 version by using, communication and signal processing tool boxes.

INTRODUCTION

Radio jamming is the (usually deliberate) transmission of radio signals that disrupt communications by decreasing the signal-to-noise ratio.[1] Unintentional jamming occurs when an operator transmits on a busy frequency without first checking whether it is in use, or without being able to hear stations using the frequency. Another form of unintentional jamming occurs when equipment accidentally radiates a signal, such as a cable television plant that accidentally emits on an aircraft emergency frequency. The concept can be used in wireless data networks to disrupt information flow. Itis a common form of censorship in totalitarian countries, in order to prevent foreign radio stations in border areas from reaching the country. As an

ISSN: 2320-1363

exampple, Islamic regime in Iran has been using the radio jamming to block free information across the large cities and the capital, TehranDistinction between "jamming" and "interference"Originally the terms were used interchangeably but nowadays most radio users use the term "jamming" to describe the deliberate use of radio noise or signals in an attempt to disrupt communications (or prevent listening to broadcasts) whereas the term "interference" is used to describe unintentional forms of disruption (which are far more common). However the distinction is still not universally applied. For inadvertent disruptions, see electromagnetic compatibility.

Method

Intentional communications jamming is usually aimed at radio signals to disrupt control of a battle. A transmitter, tuned to the same frequency as the opponents' receiving equipment and with the same type of modulation, can, with enough power, override any signal at the receiver. Digital wireless jamming for signals such as Bluetooth and WiFi is possible with very low power. The most common types of this form of signal jamming are random noise, random pulse, stepped tones, warbler, random keyed modulated CW, tone, rotary, pulse, spark, recorded sounds, gulls, and sweep-through. These can be divided into two groups – obvious and subtle. Obvious jamming is easy to detect because it can be heard on the receiving equipment. It usually is some type of noise such as stepped tones (bagpipes), random-keyed code, pulses, music (often distorted), erratically warbling tones, highly distorted speech, random noise (hiss) and recorded sounds. Various combinations of these methods may be used often accompanied by regular morse identification signal to enable individual transmitters to be identified in order to assess their effectiveness. For example, China, which used jamming extensively and still does, plays a loop of traditional Chinese music while it is jamming channels (c.f. Attempted jamming of number stations). The purpose of this type of jamming is to block reception of transmitted signals and to cause a nuisance to the receiving operator. One early Soviet attempt at jamming western broadcasters used the noise from the diesel generator that was powering the jamming transmitter. Subtle jamming is jamming during which no sound is heard on the receiving equipment. The radio does not receive incoming signals yet everything seems superficially normal to the operator. These are often technical attacks on modern equipment, such as "squelch capture". Thanks to the FM capture effect, frequency modulated broadcasts may be jammed, unnoticed, by a simple unmodulated carrier. The receiver locks onto the larger carrier signal and hence will ignore the FM signal with information. Digital signals use complex modulation techniques such as QPSK. These signals are very robust in the presence of interfering signals. However the signal relies on hand shaking between the transmitter and receiver to identify and determine security settings and method of high level transmission. If the jamming device sends initiation data packets the receiver will begin its state machine to establish two way data transmission. A jammer will loop back to the beginning instead of completing the handshake. This method jams the receiver in an infinite loop where it keeps trying to initiate a connection but never completes it, which effectively blocks all legitimate communication.

ISSN: 2320-1363

Bluetooth and other consumer radio protocols have built in detectors so that they only transmit when the channel is free. Simple continuous transmission on a given channel will continuously stop a transmitter transmitting, hence jamming the receiver from ever hearing from its intended transmitter. Wireless networks are built upon a shared medium, making them vulnerable to jamming attacks. Such attacks are accomplished by generating intentional RF interference that does not adhere to the conventions of an underlying MAC protocol [1]. Jamming signals interfere with the transmissions of legitimate transmitters at the receiver in the sense that the signals collide and render the originally transmitted data signals uninterpretable. In contrast to traditional security primitives such as authentication, confidentiality, or integrity that can be addressed with cryptographic techniques, jamming attacks targeting the availability cannot be fended off entirely by conventional security mechanisms. While spread spectrum communication techniques are able to mitigate the effect of narrowband interference, a jammer can always disturb the communication by emitting broadband signals that exceed the power of legitimate signals. Jammers may employ a wide range of strategies to disturb wireless communications [1]–[5]. Among these existing strategies, reactive jammers that become active upon detection of transmissions over the channel have been shown not only to be the hardest to detect, but also the most energy-efficient approach, making them a serious threat in wireless networks.

In addition, recent work [6] has demonstrated that reactive jammers can be implemented on inexpensive commercial offthe- shelf (COTS) platforms such as the USRP from Ettus Research, and that reactive jamming can be triggered selectively, for example, on any field of the packet header, making it a realistic threat for wireless communications.

Fundamentally, jamming cannot be prevented by design, hence it is important to understand how it works and, in turn, how to detect its presence. This paper proposes a novel method to detect reactive jammers in wireless communication systems.

The key idea is to use information extracted from the first few jamming-free bits received during the signal synchronization phase of regular packet reception to discriminate jammed packets from packets that are lost due to natural causes such as bad channel conditions. This problem is known to be challenging in real-world environments [1], [7].

Our work targets direct sequence spread spectrum (DSSS) communication systems such as the one employed in the IEEE 802.15.4 standard. We take advantage of the fact that the first few jamming-free bits are known a priori because they constitute a fixed preamble intended for signal synchronization at the DSSS receiver. Since the packet preamble represents the start of the DSSS signal on the air, it is unlikely that a reactive jammer jams this part of the communication because it demands very high reactivity, low signal propagation delays, and it prevents a jammer from making smart jamming decisions according to physical, MAC, or payload based rules.

ISSN: 2320-1363

EXISTING SYSTEM

TRANSMISSION AND RECEPTION MODEL

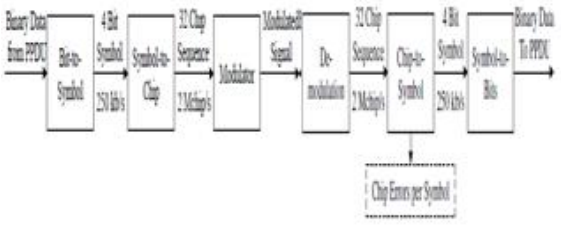

Fig.1 Communication model used

A. BACKGROUND ON IEEE 802.15.4

Packet transmission. Our work on jamming detection focuses on direct sequence spread spectrum (DSSS) communication systems, and is practically demonstrated for the 2.4GHz physical layer (PHY) of the IEEE 802.15.4 standard [8, Section 6.5]. This PHY defines a 16-ary quasi-orthogonal DSSS modulation technique; the modulation spreads a lowrate bit sequence to a higher-rate sequence, consisting of so-called chips, in the following way: binary source data is divided into groups of 4 bits (referred to as symbols) and mapped to a quasi-orthogonal 32-chip pseudo-noise sequence (b0, b1, b2, b3) → (c0, c1, . . . , c31), resulting in a chip rate of 2MChips/s (as shown in Figure 1). The effect of this spreading is an increased robustness against fading and inband interference: DSSS systems can tolerate a certain number of chip errors and still receive symbols correctly. Our proposed detection scheme relies on an estimation of the PDR based on the observation of the packet preamble. The preamble in IEEE 802.15.4 is a sequence of eight symbols 0 with the same modulation as the following data bits of the packet. After the preamble follows a start of frame delimiter (SFD; symbols 7 and 10), a frame length field indicating the duration of the frame, and finally the MAC protocol data unit (MPDU). The MPDU contains a MAC header, data payload, and ends with a frame check sequence (FCS) used to detect transmission errors. IEEE 802.15.4 does not mandate the use of error correction mechanisms, and any received packet with an incorrect FCS is hence discarded. This implies that reactive jammers can drop packets very efficiently by destroying only one or two symbols in a packet [6]. Packet reception. To receive a packet, the receiver first synchronizes with the preamble sequence to detect the symbol boundaries, i.e., the time instants when chip sequences start, and the carrier and baseband phase offsets. This timing information is subsequently used to detect the SFD and frame length field. The rest of the signal is decoded using a correlator to map each received block of 32 chips back to symbols. It is compared to the 16 predefined chip sequences Ci, i =0, 1, . . . , 15. The received chip sequence R may contain errors caused by fading or interference. The receiver chooses the best match, i.e., the Ci for which h(R,Ci) is minimal, where h(·, ·) is the Hamming distance (number of positions containing differing chips) between the two arguments. However, if too many chips are flipped (e.g., when a jammer is active), then the expression h(R,Ci) may be minimal for the wrong chip sequence Ci and the receiver interprets the chip sequence as a wrong symbol. the packet is lost due to bad channel conditions, while in the second case the packet is transmitted successfully despite chip errors. In (a), the sender starts to transmit the preamble sequence, the SFD, and the corresponding length field and MPDU (denoted here as rest of packet). During the transmission of the eight preamble symbols of the first packet, P1,2,

ISSN: 2320-1363

P1,3, P1,4 are not decoded correctly due to a high number of chip errors. In contrast, P1,7 is transmitted successfully because, as shown in (d), only three chips are flipped during the transmission and the maximum error threshold to discriminate between a correct and wrong preamble symbol is not exceeded. Finally, due to a corrupted symbol in SFD1, the synchronization of the first packet fails and the receiver is not able to decode this packet entirely. Specifically, this means that the packet is not counted as a packet error because the receiver never enters its reception mode and its FCS is not checked, making it hard to derive statistics for jamming detection when synchronization fails.

Contrary to the first packet, the second packet is transmitted successfully (c) because only the preamble symbols P2,1 and P2,5 are not correctly decoded, which allows the receiver to synchronize to the packet and decode a valid SFD. Concluding, symbol level analysis can only distinguish between symbols above and below the chip-error threshold. Instead, chip errors provide a richer information about the status of the channel and its expected PDR. In Section III, we will show that the number of chip errors is highly correlated with the probability of successful packet reception, and that the use of the preamble enables us to accurately derive PDR statistics even if a receiver never enters reception mode.

Attacker Model

We consider jammers that aim to block the entire communication over a link by emitting interference reactively when they detect packets over the air. The jammers minimize their jamming activity to only a few symbols per packet and use minimal but sufficient power to remain undetected. We assume that the jammer is able to sniff any symbol of the packet over the air in real-time and react with a jamming signal that flips selected symbols at the receiver with high probability. An attacker may therefore pursue different reactive jamming strategies [6]. It may jam (i) the MPDU, (ii) the frame length field, (iii) the SFD, or (iv) the preamble of the packet. Figure 3 illustrates jamming strategy (iii) that targets the SFD. The first two strategies cause packet losses because of resulting FCS errors, while the last two strategies introduce synchronization failures, causing the entire packet to be missed by the receiver. Such synchronization errors make it hard to devise jamming detectors because often the packet error count is used to distinguish jammed and non-jammed situations [1], [7], which cannot be derived in this situation. The experimental evaluation shows that our CER-based approach does not suffer from this restriction, and we are able to detect all four jamming strategies.

We also assume that the attacker cannot destroy all preamble symbols, i.e., at least a few symbols across several packets are available as input to our detector. We denote the time difference between the arrival of the original signal and the jammer signal at the receiver as the jamming reaction time τ. The minimal reaction time τ_{min} is bounded by the sum of the signal propagation delay between sender and jammer, the reaction delay of the jammer to process the incoming signal and to make a jamming decision, and the signal propagation delay between jammer and receiver. It is therefore safe to assume that the minimum reaction time τ_{min} is greater than the duration of one symbol (e.g., 16 μs in IEEE 802.15.4).

ISSN: 2320-1363

Otherwise it would not be possible to assess the channel state prior to jamming, i.e., not be reactive. In fact, [6] showed that the reaction time of a realistic jamming system is significantly larger than this minimum reaction delay because of the inherent hard- and software delays to detect, demodulate, process, and trigger jamming signals according to particular jamming rules. While it might be technically feasible to implement reactive devices with lower reaction delays than the duration of one symbol (for example, by using simple power detectors with analog parts [9], [10]), reactive jammers of that kind are unable to use the semantics of the signals to perform smart jamming decisions such as jamming selected packets according to specific rules (e.g., matching packet modulation or header properties).

In each experiment run, 40,000 packets of 26 bytes length are sent during 40 seconds from the transmitter to the receiver at constant rate. Varying link conditions in the cable and static experiments are obtained by adjusting the transmit power and by changing the nodes' positions. The true PDR at time t is calculated by averaging the number of correctly received packets in a window of 100 packets centered around t. This window size ensures that the true PDR is calculated over a time window smaller than the channel coherence time when moving the receiver at maximum v = 1cm/s and at a frequency of 2.4 GHz.1 Note that the mobility experiments have a relatively low node speed for the sake of determining the true PDR. We intentionally kept the node mobility low such that the channel coherence time is larger than the window size of 100 packets that is used to calculate the true PDR. Our results are thus relatively conservative with respect to mobility. As a jammer, we use the reactive jammer from Wilhelm et al. [6], which runs on the USRP2 software radio platform from Ettus Research. It can be configured to jam according to strategies (i) to (iv) introduced in Section II-B. The detection and decision logic are implemented on the FPGA of the USRP2, resulting in a minimal reaction delay of $\tau_{min} = 19\mu s$.

The key question we strive to answer is how well these metrics are able to predict the actual PDR. An important remark for the computation of the CER is the following. If too many chips are flipped, the expression h(R,Ci) is minimal for the wrong chip sequence Ci, such that the receiver interprets the chip sequence as a wrong symbol. The result is that the symbol is discarded and ignored in the computation of the average CER. This means that only a (potentially small) subset of preamble symbols is used in the estimation.

We measure the correlation of these four metrics with the PDR in various settings (cable, static, and mobile) and determine the Pearson correlation coefficient. This coefficient is an indicator of the linear correlation of two variables, where values close to zero indicate a low correlation and absolute values close to one represent a high linear dependence of two variables.

The correlations are plotted individually in Figure 4 for cable, static and mobile experiments. Since the environment has apparently only little impact on the distribution of the metrics, we compute a single correlation coefficient over all three environments for each metric in the further analysis. The best correlation is achieved for the CER metric (Figure 4(c)) with an absolute correlation coefficient of 0.965, followed by the SNR (Figure 4(d)) with an

ISSN: 2320-1363

absolute correlation coefficient of 0.92. The other two metrics perform significantly worse.

The number of decoded preamble symbols per successfully delivery packet (Figure 4(a)) achieves an absolute coefficient of only 0.559, while the number of consecutively decoded preamble symbols per transmitted packet (Figure 4(b)) exhibits an absolute correlation coefficient of 0.762.

Given the lower correlation of the two symbol error-based metrics, we do not consider these any further and focus in the following on the most promising two: the CER and SNR based metrics. As a next step, we analyze the correlation coefficient over different time intervals, i.e., when the metrics are averaged over varying window sizes. Small window sizes are considered particularly important when the jamming detection algorithm is expected to perform fast. Figure 5 shows how the absolute value of the correlation coefficient of the CER and SNR-based metrics varies with the number of packets used for computing these metrics. As we can see the correlation is dependent on the window size. However, for any fixed window size, the CER-based metric outperforms the SNR-based one. We therefore conclude that the number of chip errors in the preamble is the best metric among those considered.

In a first step, we estimate the instantaneous (per-packet) PDR after the reception of the preamble of packet k as

$$PDR_{\text{inst}}(k) = g\left(\frac{\sum_{i=1}^{32}\sum_{j=1}^{|\mathcal{S}_k|}(P_{k,j}[i] \oplus P[i])}{|\mathcal{S}_k|}\right)$$

where Pk,j [i] is a vector containing the 32 chips of the j-th received preamble symbol of packet k for i = 1, 2, . . . , 32, P[i] denotes a vector with the expected chips of the known preamble symbol, ⊕ is the exclusive OR operator, and |Sk| is the number of received preamble symbols for packet k. The function g(·) models the empirical distribution of the PDR versus CER as shown in Figure 4(c). For best results, we use a polynomial regression function. We have experimented with polynomials of different degrees. The root mean square error of the fit could significantly be decreased up to a fifth degree polynomial. Higher degrees only resulted in minimal improvements. The fifth degree polynomial we used in this paper is of the form

$$g(p) = a_5\, p^5 + a_4\, p^4 + a_3\, p^3 + a_2\, p^2 + a_1\, p + a_0$$

with the parameters of the fit being a5 = 0.016, a4 = −0.33, a3 = 2.41, a2 = −7.26, a1 = 8.83, a0 = −3.24. The root mean square error for this polynomial regression function is below 3% across the entire range.

While PDRinst(k) provides a very fast estimate of the link quality, it is subject to large fluctuations as shown in Figure 6(a). The figure compares the fluctuation of the instantaneous PDR on a static link to the true PDR defined as the ratio of correctly received packets to the total number of sent packets for a fixed time window of 100 packets

CER-BASED JAMMING DETECTION

In this section, we describe our jamming detection scheme that relies on the packet delivery model introduced in the previous section. The basic idea is that the receiver computes two metrics based on the incoming traffic, the observed and an

ISSN: 2320-1363

estimated PDR. Observed PDR. The observed packet delivery ratio PDRo(t) at time t is calculated by counting the ratio of correctly received packets over the total number of transmitted packets in a sliding observation window:

$$PDR_o(t) = \frac{\#\text{ of correct packets in } [t-W,t]}{\#\text{ of transmitted packets in } [t-W,t]}$$

To determine the number of correctly received packets, the receiver checks the FCS of all received packets and, if correct, increments a counter. Determining the total number of transmitted packets at the receiver must take into account that a reactive jammer might successfully jam all SFDs of the transmitted packets, thus preventing any successful packet synchronization at the receiver. The only reliable information source is therefore within the preamble since the reactive jammer is not capable to jam all the preamble symbols.

Therefore, the receiver counts the received preamble symbols and increments its counter of transmitted packets when at least one symbol 0 is detected within a sliding time window of the size of the preamble. Note that when facing an extremely fast reactive jammer, i.e., one that jams close to the sender on any power elevation over the channel without attempting to decode the preamble signals, our method might still detect 0 symbols in the payload of packets. We do not attempt to discriminate those symbols from the preamble symbols as they are still useful to estimate the PDR. In this case, the attacker would be forced to fully destroy a packet to erase all 0 symbols to mitigate our jamming detection mechanism, which greatly sacrifices the energy and stealth benefits of reactive jamming.

The observed PDRo should be calculated over a time window shorter than the channel coherence time, but sufficiently long to capture enough packets to derive a statistically relevant average. We have experimented with different values in the cable, static and mobile environments. A window size of around 100 data packets has proved to be a good choice across all environments, while not being highly sensitive to variations of this parameter. Hence, in this paper, we use a fixed window size of W = 100 ms, corresponding to roughly 100 data packets at the actual transmission rate of the sender.

Estimated PDR. The second metric is an estimated PDR based on the CER metric. As shown in Figure 8, the IEEE 802.15.4 receiver demodulates an incoming signal and attempts to map each demodulated 32-chip sequence to a known symbol. When the receiver is not synchronized yet, it attempts to map the incoming sequences to symbol 0. This is done with hard-decision decoding, that is, the receiver checks if the Hamming distance of the received chip sequence is smaller than a threshold value. This threshold value (4 for our receiver) is usually significantly below the mean Hamming distance of the symbols to prevent the receiver to synchronize on noise. To calculate a statistically relevant CER, the receiver averages the Hamming distances of multiple preamble symbols. We stress that the calculated average is not constrained to include only preamble symbols from a single packet. For example, when a jammer is reacting very quickly and jams symbols at positions 2 to 8 in the preamble, the received chip sequences 2 to 8 are not accounted for

ISSN: 2320-1363

the statistics because, due to chip flipping, their Hamming distance becomes larger than the hard decoding threshold and these symbols are hence not interpreted as 0. Similarly, when the link conditions are poor, a receiver might miss multiple symbols in a preamble. However, we do not require to detect any other field of the packet like the SFD or FCS, enabling our approach to detect a broader range of jamming attacks.

After receiving enough 0 symbols, the estimated PDR is calculated as Jamming detection. We define a heuristic hypothesis test based on the relative difference Δ between the estimated and observed PDR

$$\Delta = \frac{PDR_e - PDR_o}{PDR_e}.$$

Let us define the null hypothesis H0 and the alternative

hypothesis H1 as

H0 :"Normal transmission,"

H1 :"Jammed transmission."

Then the test is as follows:

accept H1, if Δ > ,

stay with H0, if Δ ≤ ,

where represents a tolerance level that directly affects the false positive and false negative detection rates. Let Λ() be the sum of the false positive and false negative detection

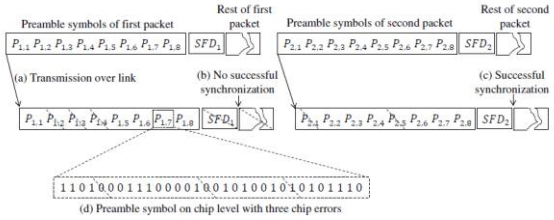

Fig 2. Packet jammed

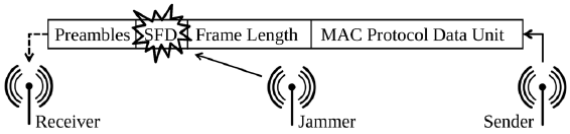

Fig.3 Jamming scenario

PROPOSED SYSTEM

Transmitter

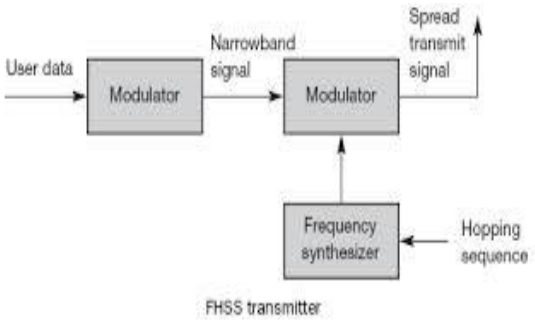

Fig.4 Transmitter part

Receiver

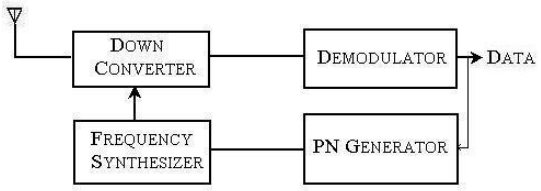

Fig.5. Receiver part

ISSN: 2320-1363

Normally binary or M-ary FSK mosulation schemes are used in FHSS. Based on the symbol transmitted , any one of the M frequencies will be used. The output signal from the modulator will be translated in frequency by an amount that is determined by the pseudo noise (PN) sequence, which in turn , is used to selsct a frequency that is synthesized by the frequency synthesizer. The frequency translated signal is mixed with the output from the FSK modulator and transmitted. If the PN generator output has m bits then 2^m-1 frequency translations are possible.

In the receiver , an identical PN generator, that is synchronised with the received signal, is used to control the output of the frequency synthesizer. By mixing the synthesizer output with the received signal, the frequency translation introduced at the transmitter can be removed. The resultant signal is demodulated by means of an FSK demodulator. A signal for maintaining synchronism of the PN generator with the frequency translated received signal is usually extracted from the received signal.

Technical considerations

The overall bandwidth required for frequency hopping is much wider than that required to transmit the same information using only one carrier frequency. However, because transmission occurs only on a small portion of this bandwidth at any given time, the effective interference bandwidth is really the same. Whilst providing no extra protection against wideband thermal noise, the frequency-hopping approach does reduce the degradation caused by narrowband interference sources.

One of the challenges of frequency-hopping systems is to synchronize the transmitter and receiver. One approach is to have a guarantee that the transmitter will use all the channels in a fixed period of time. The receiver can then find the transmitter by picking a random channel and listening for valid data on that channel. The transmitter's data is identified by a special sequence of data that is unlikely to occur over the segment of data for this channel and the segment can have a checksum for integrity and further identification. The transmitter and receiver can use fixed tables of channel sequences so that once synchronized they can maintain communication by following the table. On each channel segment, the transmitter can send its current location in the table.

In the US, FCC part 15 on unlicensed system in the 902–928 MHz and 2.4 GHz bands permits more power than non-spread-spectrum systems. Both frequency hopping and direct sequence systems can transmit at 1 Watt. The limit is increased from 1 milliwatt to 1 watt or a thousand times increase. The Federal Communications Commission (FCC) prescribes a minimum number of channels and a maximum dwell time for each channel.

In a real multipoint radio system, space allows multiple transmissions on the same frequency to be possible using multiple radios in a geographic area. This creates the possibility of system data rates that are higher than the Shannon limit for a single channel. Spread spectrum systems do not violate the Shannon limit. Spread spectrum systems rely on excess signal to noise ratios for sharing of spectrum. This property is also seen in MIMO and DSSS systems. Beam steering and directional antennas also facilitate increased system performance by providing isolation between remote radios.

ISSN: 2320-1363

Up Converter

- Up conveter is a part to convert signal up for transmission. Basically, mixer part for frequency upward conversion is called UP CONVERTER. When input signal combines LO (local oscillator) signal, RF signal is generated as much as input signal with LO signal.

Down Converter

- Down converter is a part to convert RF signal down to IF or baseband. Basically, mixer part for frequency downward is called down converter. When input signal combines LO signal, IF or baseband signal is generated as much as Input signal to LO signal. In some cases, down converter includes LNA and specifically it even includes LNA and buffer AMP in satellite equipment

BER PERFORMANCE WHILE JAMMING

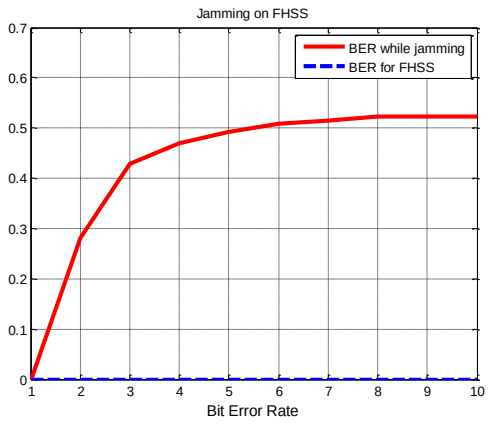

Fig 6. BER during jamming

CONCLUSION

We have proposed a novel approach to detect sophisticated reactive jamming attacks that target any part of a packet transmission in frequency hop based systems. Our approach is based on an estimation of the packet delivery probability during the signal synchronization phase of a packet transmission, which makes it suitable to detect even jammers that target the physical layer header of packets. We have analyzed the accuracy of different preamblebased metrics to predict the packet delivery probability and have shown that the chip error rate (CER) in the received preamble symbols is the most accurate estimator among the ones considered. Our experiments under real-world channel conditions have shown that it is possible to predict the PDR using the CER derived from just a few symbols in the preamble with a mean absolute estimation error of approximately 5% across all channel conditions. Based on this, we have developed a jamming detection algorithm that compares the estimated delivery probability with the observed delivery ratio to distinguish between packet losses caused by jamming and losses due to bad channel conditions. Our technique is able to detect reactive jammers that jam all packets on links with a PDR above 0.3 without any false positive or negative detection errors.

REFERENCES

[1] N. Thepvilojanapong, Y. Tobe, and K. Sezaki, "On the construction of efficient data gathering tree in wireless sensor networks," in Proc. 2005 IEEE ISCAS, pp. 648–651.

[2] C. Liu, K. Wu, and J. Pei, "An energy-efficient data collection framework for wireless sensor networks by exploiting spatiotemporal correlation," IEEE Trans. Parallel Distrib. Syst., vol. 18, no. 7, pp. 1010–1023, July 2007.

ISSN: 2320-1363

[3] Y. Wu, S. Fahmy, and N. B. Shroff, "On the construction of a maximum lifetime data gathering tree in sensor networks: NP-completeness and approximation algorithm," in Proc. 2008 IEEE INFOCOM, pp. 356–360.

[4] D. Luo, X. Zhu, X. Wu, and G. Chen, "Maximizing lifetime for the shortest path aggregation tree in wireless sensor networks," in Proc. 2011 IEEE INFOCOM, pp. 1566–1574.

[5] J. Liang, J. Wang, J. Cao, J. Chen, and M. Lu, "An efficient algorithm for constructing maximum lifetime tree for data gathering without aggregation in wireless sensor networks," in Proc. 2010 IEEE INFOCOM Mini-Conference, pp. 1–5.

[6] W. Bechkit, M. Koudil, Y. Challal, A. Bouabdallah, B. Souici, and K. Benatchba, "A new weighted shortest path tree for convergecast traffic routing in WSN," in Proc. 2012 IEEE Symposium on Computers and Communications, pp. 187–192.

[7] M. Ma and Y. Yang, "SenCar: an energy efficient data gathering mechanism for large scale multihop sensor networks," IEEE Trans. Parallel Distrib. Syst., vol. 18, no. 10, pp. 1476–1488, Oct. 2007.

[8] M. Ma and Y. Yang, "Data gathering in wireless sensor networks with mobile collectors," in Proc. 2008 IEEE International Parallel and Distributed Processing Symposium, pp. 1383–1391.