JAMMING ATTACKS AND COUNTERMEASURES IN WIRELESS PERSONAL AREA NETWORKS

LIU GUOBIN

Ph. D

The Hong Kong Polytechnic University

2012

The Hong Kong Polytechnic University
Department of Computing

Jamming Attacks and Countermeasures in Wireless Personal Area Networks

By

Guobin Liu

SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

August 11, 2011

Certificate of Originality

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

____________________ (Signature)

Guobin Liu (Name of Student)

Abstract

The jamming attack is one of the serious threats to wireless sensor networks (*WSNs*) using the IEEE 802.15.4 standard. In such an attack, jammers, who launch the attack, can dramatically degrade the network performance by interfering transmitting packets. Therefore, the study of the jamming attack and its countermeasures has become an important aspect of the WSN security.

First, we present an energy-efficient Reduction-of-Quality (*EERoQ*) attack against coordinators to prevent them from receiving normal packets from neighbors. Existing jamming attacks are not efficient because they rely on overwhelming the victim with load that constantly exceeds targets capacity. In *EERoQ*, attackers use periodic jamming signal to block the transmission of data packets or ACK frames. As a result, a sender node will enter the retransmission state continuously and the network throughput will dramatically decrease. We use potency to denote the efficiency of the attack and define potency to be the ratio of the damage caused by attackers to the cost of launching *EERoQ*. In our model, an attacker can maximize damage per unit cost by maximizing the attack potency. In *EERoQ*, the generation of jamming signal is one of the critical technologies to launch the attack. Therefore, we present a strategy for implementing jamming signal by exploiting the CSMA-CA mechanism. Our experiments on an IEEE 802.15.4 compatible sensor network show that the proposed attack strategy is an energy-efficient way to implement the RoQ attack.

Second, we present an effective dynamic jamming attack *(EDJam)* in an 802.15.4-compliant *WPAN* and use the Stackelberg game to formulate the dynamic procedure of competition between the *EDJam* attacker and defending networks. Most existing jamming attacks can cause negative interference, but the attack strategies are usually not adjusted against the countermeasures that are currently taken. In the proposed attack, a jammer who is aware of a change in the network defense strategy, e.g. the use of a dynamic retransmission mechanism, may choose a better strategy to make more damage to the network with less cost. Similarly, a well-protected network can change its defense strategy against the *EDJam*. Therefore, a Stackelberg game can precisely reflect the procedure of a dynamic jamming attack and network defense. Moreover, we derive a unique Nash Equilibrium point in analytical format in this game. Based on an equilibrium analysis, we discuss the condition under which a defense strategy will increase the utility of the network and a dynamic retransmission mechanism defense strategy is proposed accordingly. The simulation results show that *EDJam* can be more cost-efficient than continuous, random and fixed-period jamming.

Finally, we describe an energy-efficient keyless *DSSS* (*EUDSSS*) proposed for energy-constrained wireless networks. Prior research proposed various keyless spread spectrum technologies that focus on the capacity of jamming-resistant without the pre-share secret key. Their design is not energy-efficient, which make them not suitable to be applied in energy-constrained networks such as wireless sensor networks. Therefore, we proposed the *EUDSSS* to balance the capacity of jamming-resistance and overhead by maximizing the tradeoff between correct bit rate and the additional communication bits comparing with traditional *DSSS*. The simulation results show that *EUDSSS* can achieve adequate reliability with small overhead.

Keywords: wireless sensor network, 802.15.4, jamming attack, Stackelberg game, *DSSS*.

Publication

Conference Papers

1. **Guobin Liu**, Jiaqing Luo, Qingjun Xiao and Bin Xiao, “EDJam: Effective Dynamic Jamming Against IEEE 802.15.4-Compliant Wireless Personal Area Networks”, *IEEE International Conference on Communications (ICC 2011), Kyoto, Japan*, June 2011.

2. Qingjun Xiao, Bin Xiao, Jiaqing Luo and **Guobin Liu**, “Reliable Navigation of Mobile Sensors in Wireless Sensor Networks without Localization Service”, *17th IEEE International Workshop on Quality of Service (IWQoS 2009), pp. 1-9, Charleston, South Carolina*, July 13-15, 2009.

3. Jiaqing Luo, Bin Xiao, **Guobin Liu**, Qingjun Xiao and Shijie Zhou, “Modeling and analysis of self-stopping BTWorms using dynamic hit list in P2P networks”, *IEEE International Symposium on Parallel and Distributed Processing (SSN2009), pp.1-8, Rome, Italy*, May 2009.

Acknowledgement

First, I would like to thank my supervisor, Dr. Bin Xiao, for his rigorous supervision of my research. I thank him for his guidance, patience and encouragement during my PhD. study. He unremittingly trained me to be a good researcher. He taught me how to find research issues and how to solve problems. Time and again, he showed me how to express my ideas and write research papers. His knowledge, vision, passion, professional spirit and attitude towards the research deeply inspired me. Without his help and support, this body of work would not have been possible. What I have learned and experienced during the PhD. study will always help and encourage me in the future.

Second, I would like to thank another excellent and distinguished person, Dr. Qixing Wang. I would like to express my heartfelt gratitude to him for his insightful and illuminative suggestions on my research. His acuminous insight and guidance is invaluable to the accomplishment of this thesis.

Third, I would like to thank all my teachers from whom I have learned so much during my long journey in academia. They are Professor Jane You, Dr. Henry Chan, Dr. Li Jiang and Dr. Xiaofeng Fu at the Hong Kong Polytechnic University, and many others. In addition, I also wish to thank Jiaqing Luo, Qingjun Xiao and Kai Bu, who have rendered their help to my research work in one way or another and shared with me the pleasure of study at the Hong Kong Polytechnic University.

Last but not least, I thank my parents and wife for their continuous love, support, trust, and encouragement for me through the entire period of my study. Without them, none of this would have been possible.

Contents

Certificate of Originality **i**

Abstract **ii**

Publications **v**

Acknowledgements **vi**

Table of Contents **viii**

List of Figures **xiii**

List of Tables **xvii**

List of Abbreviations **xviii**

1 Introduction **1**

1.1 Jamming Attacks 1

1.2 Dynamic Jamming Attacks 4

1.3 Spread Spectrum against Jamming Attacks 6

1.4 Contributions of the Thesis 9

1.5 Outline of the Thesis 11

2 Background and Literature Review 12

2.1 Transmission and Retransmission in IEEE 802.15.4 14

2.1.1 Personal area networks (PANs) 14

2.1.2 The CSMA-CA transmission mechanism 17

2.1.3 Retransmission 19

2.2 DoS and RoQ in WSNs 22

2.3 Jamming Attacks 24

2.4 DSSS 26

2.5 Summary 28

3 EERoQ: Energy-Efficient Reduction of Quality Attack 29

3.1 Overview 29

3.2 An Energy-Efficient RoQ Attack Model 30

3.2.1 Mathematical model 31

3.2.2 Calculating Potency 33

3.3 Generating Burst Traffic 39

3.4 Experiments 43

3.4.1 Configuration 43

3.4.2 Experiment Results 45

3.5 Summary 51

4 A Stackelberg Game for Dynamic Jamming Attack and Defense 54

4.1 Overview 55

4.2 Attack Model 56

4.3 Game Between Jammers and the Network 59

4.3.1 Utility Functions 61

4.3.2 Maximizing the Attacker's Utility 64

4.3.3 Network Defense Strategy 67

4.4 Performance and Evaluation 73

4.4.1 Impact of the Jamming Period and the Retransmission Timer 73

4.4.2 Comparison 76

4.5 Summary 78

5 Energy-Efficient Uncoordinated Direct Sequence Spread Spectrum Against Jamming Attacks 81

5.1 Overview 81

5.2 Network and Jammer Models 83

5.2.1 DSSS 83

5.2.2 Jamming Attacks 86

5.2.3 Jamming Resistance 87

5.2.4 Overhead 89

5.3 A Simple Uncoordinate DSSS 89

5.3.1 A Simple Uncoordinate DSSS 90

5.3.2 Optimal Value of l_s 93

5.4 An Efficient Uncoordinate DSSS 95

5.4.1 Tradeoff Between Resistance and Overhead 95

5.4.2 Optimizing the Tradeoff 97

5.4.3 Defending Reactive Jamming 99

5.5 Simulations 101

5.5.1 Simulations Setting 101

5.5.2 Bit Error Rate 102

5.6 Summary 107

6 Conclusions and Future Works 108

6.1 Conclusions 108

6.2 Future Works 110

List of Figures

2.1 Transmission model. 15

2.2 The structure of superframe. 16

2.3 The procedure of slotted CSMA-CA algorithm. 18

2.4 A sender continuously sends out packets from $t = 0$. It successfully receives ACK at “A”, “C” and “D”. However the sender needs to retransmit at “B” because the sender does not receive a correct ACK in t_{r_0}. 22

3.1 Attack Model. 32

3.2 Experiment 1: Attacker follows CSMA-CA. 46

3.3 Experiment 2: Modify BE. 46

3.4 Experiment 3: Modify BE and CW. 47

3.5 Experiment 4: Modify BE, CW and CCA. 48

3.6 Drop rate versus packet sending frequency of attacker when APL is 2. 49

3.7 Drop rate versus packet sending frequency of attacker when APL is 4. 50

3.8 Drop rate versus packet sending frequency of attacker when APL is 8. 50

3.9 Drop rate versus packet sending frequency of attacker when APL is 12. 51

4.1 In the attack model, l is the duration of the jamming signal. T_i and T_j are the optimal jamming periods at different times under an *EDJam* attack. P is the packet drop rate during the jamming signals, which changes with the strength of the signals. 58

4.2 The procedure of the Stackelberg game for a dynamic jamming attack and network defense. 60

4.3 Network satisfactory function of the retransmission mechanism. . 63

4.4 Attacker's utility function of the jamming period. 65

4.5 Network utility (π_d) versus retransmission timer (t_r). 70

4.6 Jammers and the network's best-response curves. The intersection point is the equilibrium point. 72

4.7 Game process and the normalized throughput. 74

4.8 The optimal network retransmission timer (t_r) and jamming period (T) versus the drop rate during jamming P. 76

4.9 Comparison among EDJam, continuous jamming and random jamming. 77

4.10 Comparison between EDJam and fixed-period jamming. 79

5.1 A simple communication model of DSSS. 85

5.2 A spreading and de-spreading example in DSSS. 85

5.3 Spreading and de-spreading of *SUDSSS*. 94

5.4 Bit error rate versus spread code length. 95

5.5 The theoretical tradeoff versus the length of spread code. 98

5.6 An example to resist the reactive jamming attack. 100

5.7 Length of spreading code versus bit error rate under various jammer capabilities. 103

5.8 Comparison between theoretical and simulation results. 104

5.9 Normalized overhead versus the length of spread code. 105

5.10 The communication overhead with respect to the length of spread code. 106

5.11 The computation overhead with respect to the length of spread code. 106

5.12 The storage overhead with respect to the length of spread code. . . 107

List of Tables

3.1 Abbreviation. 44

List of Abbreviations

WSN	Wireless Sensor Network
RoQ	Reduction of Quality
EERoQ	Energy-Efficient Reduction of Quality
EDJam	Effective Dynamic Jamming Attack
DSSS	Direct Sequence Spread Spectrum
EUDSSS	Energy-Efficient Uncoordinate Direct Sequence Spread Spectrum
BE	Backoff Exponent
CW	contention Window
CCA	Clear Channel Assessment
IEEE	Institute of Electrical and Electronics Engineers
ACK	Acknowledgement
CAP	Contention Access Period
DoS	Denial-of-Service
RoQ	Reduction of Quality
CSMA-CA	Carrier Sense Multiple Access with Collision Avoidance
BER	Bit Error Rate

Chapter 1

Introduction

With widespread commercial implementation of the 802.15.4-compliant Wireless Personal Area Network (*WPAN*) in recent years, the demand for security has grown rapidly. The jamming attack is one of the security threats that can lead to great damage in the real world. Wireless communication is vulnerable to jamming attacks due to the open environment. Jammers can send out interference signals to disrupt the legitimate communication [Law and Palaniswami, 2009, Xu et al., 2005, Xu et al., 2006, Negi and Perrig, 2003, Poisel, 2006a]. By using jamming attacks, the throughput of network could dramatically decrease. To defend the jamming attack, one of the effective countermeasures is spread spectrum technique (e.g. DSSS and FHSS).

1.1 Jamming Attacks

IEEE 802.15.4 standard is widely used in Wireless Sensor Networks (*WSNs*) where it supports functions of the lower network layers of wireless personal area networks (*WPANs*) which are concerned with low-cost, low-rate, short-range ubiquitous wireless communications between devices. For example, IEEE 802.15.4 has been implemented in sensor nodes such as Telos [Polastre et al., 2005] and MicaZ [Technology,]. One common way to attack WSNs that use the IEEE 802.15.4 standard is to apply a denial-of-service (DoS) approach. Such approach may degrade network performance by attacking coordinator nodes (e.g. data sinks, cluster heads) so as to reduce the network throughput.

Attacking a coordinator involves jamming it with malicious jamming signals so that data packets or ACK frames are dropped and senders must enter the retransmission state, spending most of their time on waiting for the next opportunity to transmit. A number of attacks [Yu and Xiao, 2006, Ye et al., 2004, Zhu et al., 2004, Deng et al., 1999, Zhang et al., 2005] are presented to attack coordinator and deny its service (e.g. data aggregating). However they rely on overwhelming the victim with load that constantly exceeds target's capacity or exploit the protocol vulnerability. These attacks are not energy-efficient or, in the term used in this thesis, they have low attack potency, where potency is the ratio of the damage caused by an attack to the cost of launching it. Another disadvantage of such attacks is that they can be detected through using statistical mechanisms because of their cruel behaviors. Therefore we present an energy-efficient RoQ attack model to balance the attack effect and the cost of attack.

We present an Energy-Efficient Reduction-of-Quality (*EERoQ*) attack that prevents coordinators from receiving legitimate packets from neighbors with a high level of attack potency. Our attack is efficient because it uses periodic jamming signal to block the transmission of data packets from senders or ACK frame from receivers, causing senders to believe that the previous packet transmission failed and so disrupting the communication. Therefore, the sender will enter retransmission state constantly. Moreover, because it does not continuously jam the communication channel, it does not provoke defensive counter-measures.

To implement the jamming signal, our approach exploits the Carrier Sense Multiple Access (*CSMA-CA*) mechanism which is a contention-based access mechanism for the fair use of channel. In *CSMA-CA*, a backoff algorithm is employed to improve the fairness of channel access but which is also a vulnerability in that it allows attackers access the channel quickly by modifying proper parameters.

According to the CSMA-CA mechanism prescribed in IEEE 802.15.4, normal senders must wait for a period and confirm that the channel is idle before sending out a packet. To exploit this, we modify CSMA-CA to decrease the waiting time of attackers. This both reduces the chances of detection and gives attackers an advantage in channel contention, allowing them to generate burst traffic effectively, preventing sender from sending packets or receiving ACK frames according to a certain probability. To test our model, we implemented it in a real WSN composed of MicaZ motes and sent malicious packets using a variety of payload lengths and sending frequencies. The cost/damage tradeoff can be adjusted by using the length of a burst of traffic, quantifying potency as the ratio of the damage caused by attackers to the cost of launching an attack, where damage is defined as the reduction of throughput and cost as the ratio of energy consumption of attack to energy consumption of network.

1.2 Dynamic Jamming Attacks

With widespread commercial implementation of the 802.15.4-compliant Wireless Personal Area Network (*WPAN*) in recent years, the demand for security has grown rapidly. The jamming attack is one of the security threats that can lead to great damage in the real world. Nowadays, however, with the help of a wireless sniffer, jammers can easily obtain transmitting packets in the open wireless communication environment that will allow them to analyze changes in critical parameters and jam the channel more smartly. These parameters can reveal some configuration information (e.g. transmission and countermeasure) the network is using. Therefore, Jammers can dynamically adjust their strategy of attack after detecting the kind of environment that will make it possible for them to maximize their damage to the network, e.g. by the reduction of network throughput. Similarly, in order to fully utilize the channel bandwidth, legitimate users can dynamically change their defense strategies in response to the detection.

Most existing works on jamming attacks fall into one of the following two categories according to whether the network configuration is known by attackers.

In the first category, jammers are unaware of the network configuration. This category includes continuous, random, and deceptive jamming [Xu et al., 2006]. The random jamming is energy-efficient but less effective. Both of continuous and deceptive jamming are effective but consume a great deal of energy.

The second category assumes that jammers are aware of the network configuration so that a jammer can adopt a relevant strategy of attack. Under this category, a typical method of jamming is reactive jamming [Xu et al., 2006]. It can cause the network throughput to fall to zero or almost zero. However it is not an energy-efficient method of attack because the jammer consumes energy sooner than the victims, given comparable energy budgets. A more efficient jamming attack is proposed in [Li et al., 2007], in which the jammer controls the probability of jamming and the transmission range to cause maximal damage to the network in terms of corrupted communication links. However, the jamming transmission range can be difficult to control because the range depends on the circumstances.

In this work, we propose an effective dynamic jamming attack (*EDJam*) to efficiently corrupt the legitimate communication. The jammer adjusts the jamming period in order to achieve maximal attack utility, with more damage done to network at less cost to launch the jam. Likewise, as a defender, the network would dynamically select a retransmission mechanism to maximize its utility of high throughput and reliability. In order for the jammer to maximize its utility, it needs to know the current value of the network retransmission timer (the longest waiting time for the ACK frame). Accordingly, the network would need to know the current period of jamming. Therefore, we use a dynamic competition model to describe the procedure of attacker jamming and the network defending.

In our model, we assume that both the network and attackers are rational and selfish, in that they are interested in maximizing their own utilities. The model of attack that is being considered can be analyzed by game theory, characterized by a competition involving two players. One player (network) optimizes its strategy based on the knowledge of the effect of its decision on the behavior of another player (attacker). To study this competition procedure, we use an analytical model named the Stackelberg game [Fudenberg and Tirole, 1993]. We prove that there is a unique equilibrium point for this Stackelberg game under the following several constraints: the length of the jamming signal, the jamming period, and the power of the jamming signal.

1.3 Spread Spectrum against Jamming Attacks

Wireless communication is vulnerable to jamming attacks due to the open environment. A jammer can send out interference signals to disrupt the legitimate communication among nodes [Law and Palaniswami, 2009, Xu et al., 2005, Xu et al., 2006, Negi and Perrig, 2003, Poisel, 2006a]. Therefore, jamming-resistant communication is becoming increasingly important to modern applications (e.g. military and GPS etc.). One of the effective countermeasures to resist jamming attacks is spread-spectrum technique, such as Direct-Sequence Spread Spectrum (DSSS) and Frequency Hopping Spread Spectrum (FHSS). By using DSSS, the transmitted radio signal takes up more bandwidth than the original radio signal. FHSS is a mechanism of transmitting radio signals by switching a carrier among many frequency channels. There are two categories of spread spectrum technology: key-based and keyless.

In the key-based spread-spectrum technologies [Hang et al., 2006, Li et al., 2009, Zhan et al., 2005, Huang and Li, 2001], a pre-share secret key is required to generate sequence of spread code or hopping frequency. The pre-share secret key imposes limits on the use of common spread-spectrum techniques in scenarios where such secret key is known by jammers or cannot be pre-shared [Strasser and Pöpper, 2008, Pöpper et al., 2009, Baird et al., 2007]. First, an attacker may learn the shared key by malicious receivers or compromised nodes. If the key is known by attackers, then they can predict the sequence of spread code or hopping frequency and emit strong interference signals using that sequence, thus masking the signal sent by the legitimate user. Second, the secret key is difficult to be distributed in large scale wireless networks. Therefore, we need a keyless spread spectrum technology to overcome these limitations and provide reliable communications.

Key-less spread spectrum technologies can avoid the pre-distribution of the secret key [Liu et al., 2010, Strasser and Pöpper, 2008, Pöpper et al., 2009, Baird et al., 2007]. The pioneer research on key-less spread spectrum technologies have been proposed in [Baird et al., 2007] where concurrent codes in combination with UWB pulse transmissions are used. In [Pöpper et al., 2009], authors propose a solution called Uncoordinated DSSS (UDSSS) that enables spread-spectrum jamming-resistant broadcast communication without the pre-shared key. In this work, we study a much less explored function that is very useful in energy-constrained wireless networks. Consider a wireless network (e.g. wireless sensor networks) which is composed of battery-operated nodes (e.g. MicaZ) and batteries are difficult for replacement (e.g. in battle field). If some batteries are exhausted, the network could crash (e.g. the routing protocol is not maintainable when these nodes can no longer forward routing messages). Meanwhile, the network needs to apply the spread spectrum technology to guarantee the reliability of communications. However, existing keyless DSSS technologies do not consider from the angle of energy-efficiency. A keyless DSSS is essential to achieve energy-efficiency with adequate reliability of communication.

The capacity of jamming-resistance should not be the sole performance consideration. The energy cost may be an even more critical concern if battery-operated devices are used in wireless networks. Due to limited energy, these devices are mostly used in low-cost networks such as 802.15.4-compliant wireless personal networks. For future home automation that control by these low-cost, short-range and low-cost wireless personal networks, battery-operated devices are likely to the choice. Battery-operated devices use their own power to transmit and route. They are powered by batteries and hide in every corner of the home to monitor the environment and control equipment. Considering that devices deployed in all houses are used to set up a large network to monitor the whole building together with devices deployed outside of the house (e.g. stair and veranda etc.), recharging them is a difficult laborious operation. To prolong the devices' lifetime and reduce the frequency of battery recharge, the communication between them should be made energy-efficient. Due to scalability and dynamic of the monitor network, the requirement of pre-share key is impracticable. Meanwhile, the spread spectrum technology is necessary to guarantee the reliable communication when this network is used to monitor some dangerous events (e.g. fire) and broadcast the warning. Therefore, energy-efficient solution for keyless DSSS is required and we believe this work is the first to study the solution.

1.4 Contributions of the Thesis

This thesis makes the following contributions towards efficient, dynamic jamming and efficient countermeasure.

Energy-Efficient Reduction of Quality Attack: We propose an energy-efficient reduction of quality attack to efficiently reduce the network throughput by periodic jamming signals. To achieve the goal of high efficiency, we formulate and optimize the potency of attackers to calculate the proper period of jamming signal. Meanwhile, we present a practical strategy to generate jamming signal and implement such attack in a real wireless sensor network composed of MicaZ nodes. Experimental results validate the effectiveness and efficiency of the proposed approach.

Effective Dynamic Jamming Attack: A novel, effective dynamic jamming attack (*EDJam*) and defense model is proposed to describe the procedure of jamming and defending. Different from previous work, our model can describe the procedure of the attacker jamming and attackers and the network defending. In this procedure, they can revise their respective strategies when they detect the strategy that their rival is using. We formulate the above model as a Stackelberg game, with the network acting as the leader and the attacker acting as the follower. We prove that there exists a unique Nash Equilibrium for the game. From the analytical results, we modify the retransmission mechanism for IEEE 802.15.4 [IEEE, 2003] to defend *EDJam*. The simulation results show that the network can achieve good performance in terms of throughput and high reliability, when using the dynamic retransmission mechanism.

Efficient Uncoordinate Direct Sequence Spread Spectrum: We make two major contributions in our work. First, we propose a simple keyless *DSSS* algorithm called *SUDSSS*. The contribution of *SUDSSS*, compared with existing keyless *DSSS* technologies, is to derive the optimal length of spread code that can minimize the bit error rate in the presence of jamming attacks. Meanwhile, the proposed *SUDSSS* greatly reduce the computational overhead. Our solution keeps the spread and de-spread operation very simple, which is important for practical wireless networks. Second, we provide an energy-efficient solution for keyless *DSSS* based on *SUDSSS*. We are probably the first to consider energy efficiency for keyless *DSSS*. We reveal a fundamental tradeoff between correct bit rate and overhead. The solution achieves energy-efficiency by maximizing the tradeoff. Higher correct bit rate can be achieved at the expense of larger overhead, or vice versa. Although the jamming-resistance mechanisms in [Strasser and Pöpper, 2008, Liu et al., 2010] can provide a reliable communication (high correct bit rate), the overheads are unacceptable in the energy-constrained network. In our solution, we choose a proper parameter value that result in far smaller energy cost as well as lower bit error rate.

1.5 Outline of the Thesis

The remainder of this thesis is organized as follows. In Chapter 2, we briefly discuss the previous work on jamming attacks and countermeasures. In Chapter 3, we present how to generate jamming signals in 802.15.4-compliant wireless network. A Stackelberg game for dynamic jamming attacks is described in Chapter 4. An efficient keyless DSSS is proposed in Chapter 5. In Chapter 6, we present our conclusions and discuss future work.

Chapter 2

Background and Literature Review

In the past decade, *WSNs* experienced from academic research to practical application. They are widely used in more and more applications (home automation, environmental monitoring, industrial monitoring and control, military, security, traffic control and health care etc. [Kuorilehto et al., 2009]). A typical *WSN* is composed of a large number of small and battery-operated nodes. Generally, these nodes are randomly distributed and deployed for the collection of information. IEEE 802.15.4 Low-Rate Wireless Personal Area Network (*LR-WPAN*) together with *ZigBee* are one of the most promising technologies enabling *WSNs*. In a basic *WSN* scenario, resource constraint, wireless communication, security-sensitive data, uncontrollable environment and even distributed deployment are all vulnerabilities. These vulnerabilities make *WSNs* suffer from an amazing number of security threats. *WSNs* can only be used in the critical application after the potential security threats are eliminated [Zhang and Kitsos, 2009]. There are two types of *WSN* security threats, passive attacks and active attacks. Passive attacks can either result in the disclosure of message contents (eavesdropping) or successful *traffic analysis* [Stallings, 1995]. In *WSNs*, interception attacks form the group of passive attacks, which can be implemented by collecting messages exchanged between nodes. In *WSNs*, active attacks include modification and fabrication of information and interruption in its various forms [Stallings, 1995].

The jamming attack is one of the most serious security threats in the field of *WSNs*, which can easily make great damage to *WSNs* that utilize strong high-layer security mechanisms. Jamming is defined as the emission of radio signals aiming at disturbing the transceivers' operation [Adamy and Adamy, 2004]. In the context of *WSNs*, jamming is the type of attack which interferes with the radio frequencies used by network nodes [Shi and Perrig, 2002]. Hence, it should be classified as an active attack.

A spread spectrum is the conventional techniques to defend against jamming attacks. This technique decreases the potential interference to other receivers while achieving privacy. Spread spectrum generally employ a noise-like bit stream to spread the original narrowband information signal over a relatively wideband (radio) band of frequencies. The receiver de-spread the received signals to retrieve the original information signal. There were two motivations of spread spectrum technologies: either to resist enemy efforts to jam the communications, or to hide the fact that communication was even taking place. However, it is too energy-consuming to be widely deployed in resource-constrained wireless networks such as 802.15.4-compliant WSNs. Therefore, design of an energy-efficient spread spectrum technology is very critical in WSNs.

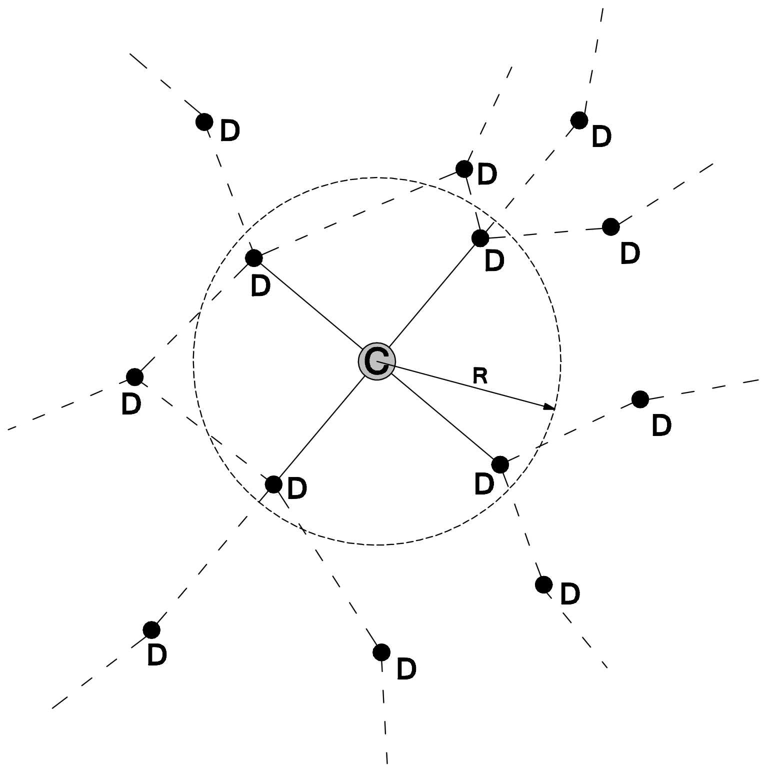
2.1 Transmission and Retransmission in IEEE 802.15.4

In this section, we present an overview of transmission and retransmission mechanisms in IEEE 802.15.4. We will exploit these two mechanisms to launch the *EDJam* attack. In this chapter, we focus on beacon-enabled uplink communication with acknowledgment. For simplicity, we discuss our attack in a single cluster using star topology.

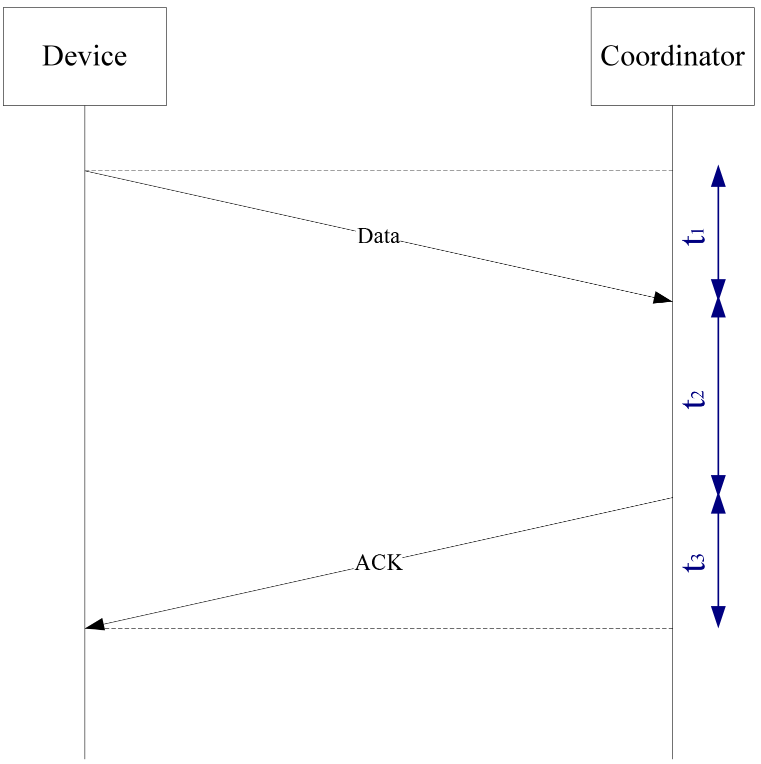
Transmission and retransmission in IEEE 802.15.4 take places within a personal area network PAN. In the following, we first describe the types of nodes, transfer model, beacon and superframe by which a PAN operates and then in turn describes the two sensor node mechanisms that our proposed *RoQ* attack model exploits to generate malicious burst traffic. These mechanisms are the transmission or channel access mechanism, Carrier Sense Multiple Access with Collision Avoidance (*CSMA-CA*) and the IEEE 802.15.4 retransmission mechanism.

2.1.1 Personal area networks (PANs)

Transmission in IEEE 802.15.4 involves protocols and an interconnection of devices through radio communication in a personal area network (PAN) made up of nodes. There are three types of nodes defined in IEEE 802.15.4: devices, coordi-



(a) Topology (D: Device, C: Coordinator).



(b) Transfer data from device to coordinator.

Figure 2.1: Transmission model.

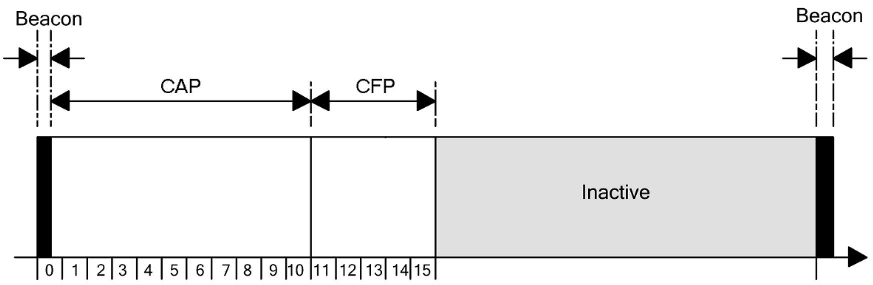


Figure 2.2: The structure of superframe.

nators, and PAN coordinators. Devices collect circumstance data and transfer the data to the coordinator according to the protocol. The coordinator acts as a cluster head, controlling its own cluster and aggregating data received from devices. The PAN coordinator is the primary controller of the PAN, receiving data from coordinators and managing all the cluster heads in the network.

Data in a PAN is transferred from a device to a coordinator; from a coordinator to a device; and between devices. In data aggregation between a device and coordinator, devices first send data to a neighbor of the PAN coordinator, and the neighbor then forwards the data to the coordinator.

The particular type of PAN that we consider in this chapter is a beacon-enabled 802.15.4 WSN that uses a superframe to synchronize the network. The superframe is responsible for allocating time slots to different functions. Figure. 2.2 illustrates a superframe structure that is defined and sent by the coordinator. As shown in Figure. 2.2, a superframe is divided into 16 slots of equal size and it begins with a network beacon. The beacon frame is transmitted in the first slot of each superframe and is used to synchronize end-devices, to identify the PAN, and to notify the structure of the superframe. The rest of the superframe is divided into a Contention Access Period (*CAP*) and Contention Free Period (*CFP*). Any device wishing to communicate during the *CAP* between two beacons shall compete with other devices that also use the slotted *CSMA-CA* mechanism to access the channel.

2.1.2 The CSMA-CA transmission mechanism

A contention-based channel access mechanism, *CSMA-CA*, is used in beacon-enabled *PANs* during the *CAP*. Following is a brief description of *CSMA-CA*: while a device wishes to transmit data frames during the *CAP*, the boundary of the next backoff slot is located and then the device waits for a random number of backoffs. If the channel is detected to be idle twice in a row, the device will commence transmitting on the next available slot boundary. If the channel is busy, following this random backoff the device waits for another random number of backoff slots before trying once again to access the channel.

In particular, using the slotted *CSMA-CA* access mechanism (2.3), a device initiates three variables: CW, *BE* and NB. CW is the contention window length. It defines the number of backoff periods that needs to be clear before the transmission can start. It is initialized to 2 before each transmission attempt and will be reset to 2 when the channel is detected to be busy. *BE* is the backoff exponent,

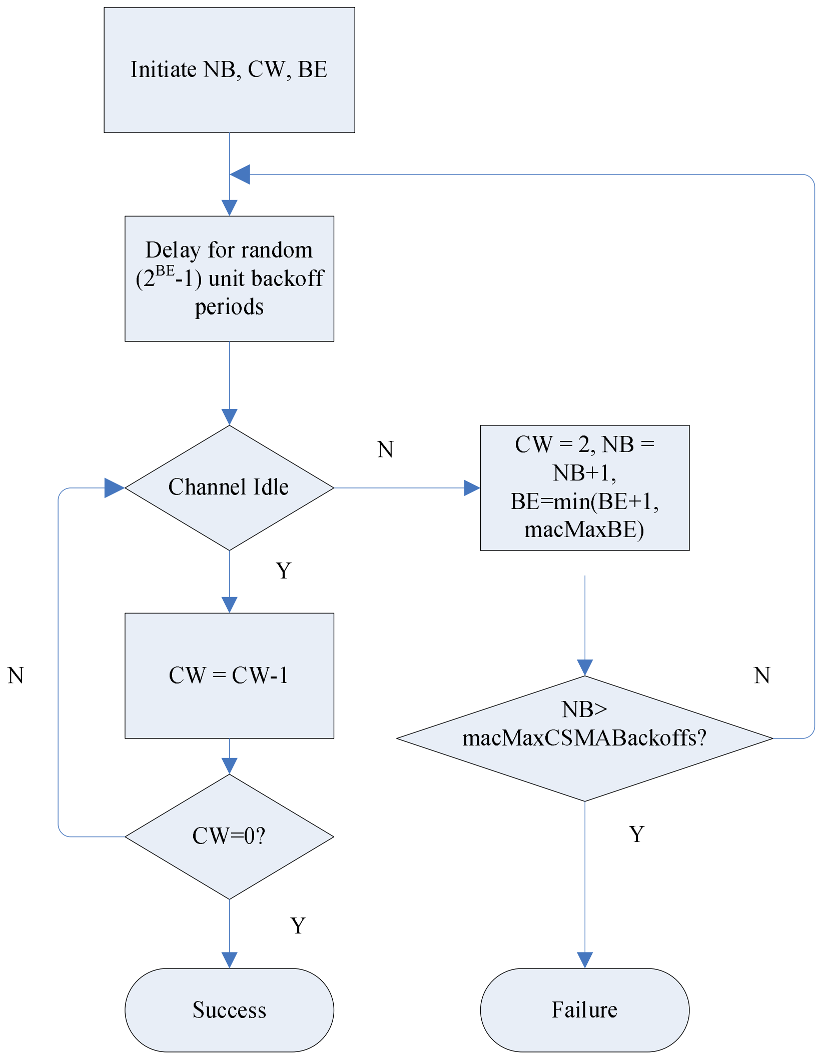


Figure 2.3: The procedure of slotted CSMA-CA algorithm.

which represents how many backoff periods a device should wait for before attempting to assess the channel. It is initialized to the minimum of two or the value of *macMinBE* (default value is 3). NB is the number of times the *CSMA-CA* algorithm was required to backoff while attempting the current transmission. It should be initialized to zero when a device attempts to commence a new transmission.

The *CSMA-CA* algorithm goes through the following steps until NB is bigger than $macMaxCSMABackoffs$ and returns *FAILURE*. First, a device waits for random $(2^{BE} - 1)$ unit backoff periods[2.1]. Then if the channel is found to be idle and CW is bigger than 0, CW decreases by 1. Otherwise, let CW be 2, increase NB by 1 and let BE be the smaller one of $BE + 1$ and $macMaxBE$. At last, when CW is equal to 0, it returns SUCCESS. In Section 3.3, we will implement the jamming signal, which is one of the critical technologies for launching our *EDJam* attack, by exploiting this *CSMA-CA* mechanism.

2.1.3 Retransmission

In the open wireless circumstance, there are many factors (noise, collision etc.) that would lead to the loss of data. Therefore, a reliable retransmission mechanism is necessary for *WPANs*. With this mechanism, a sender will retransmit a packet several times, if the sender does not receive the correct ACK frame in time or receives an incorrect ACK frame.

[2.1] One backoff period is equal to 20 symbols.

In IEEE 802.15.4, the transmitted data or MAC command frame will be acknowledged if the acknowledgment request subfield of its frame control field is set to one. If the destination receives the frame correctly, it will generate and send back an acknowledgement that contains the expected cluster identifier, Data Sequence Number (*DSN*) and source endpoint. Figure 2.1(b) illustrates the case of a successful transmission with acknowledgement. In this chapter, device and sender, coordinator and receiver, are exchangeable. A device sends a data packet to the coordinator. The transmission procedure costs t_1 time and the coordinator uses t_2 time to handle the packet. Then, the coordinator replies with an ACK frame to the device. The transmission of the ACK frame needs t_3 time, which is close to t_1.

macAckWaitDuration is the maximum number of symbols to wait for an acknowledgment frame to arrive following a transmitted data frame. It is dependent on a combination of constants, which is shown in Eq. 2.1.1. In this chapter, we denote the value of *macAckWaitDuration* as t_{r_0} and name it retransmission timer. The unit of this constant is a symbol that has a default value of 16 $microseconds$.

$$\begin{aligned} t_{r_0} = {} & aUnitBackoffPeriod + aTurnaroundTime + \\ & phySHRDuration + \lceil 6 \cdot phySymbolsPerOctet \rceil \end{aligned} \tag{2.1.1}$$

aTurnaroundTime is the maximum turnaround time from receiving mode to transmission mode (or vice versa) for a device. Its default value is 12 (in symbol periods). *aUnitBackoffPeriod* is the number of symbols forming the basic time period used by the *CSMA-CA* algorithm. Its value is 20. *phySHRDuration* is the duration of the synchronization header (SHR) in symbols. Its value is 3, 7, 10 or 40 in the standard. *phySymbolsPerOctet* is the number of symbols per octet. Its value is 0.4, 1.6, 2 or 8. They are all constants. Hence, the value of t_{r_0} is constant too.

In Figure 2.4, we use some examples to illustrate the retransmission process. If the expected acknowledgement is received within t_{r_0} symbols after the original data frame (A, C and D in Figure 2.4), the sender will believe that the transmission is successful and transmit the next packet. If the sender does not receive the acknowledgement within t_{r_0} symbols (B in Figure 2.4) or receive an acknowledgement that has a DSN, cluster identifier or source endpoint different from that of the original transmitted frame, the device will conclude that the transmission has failed. In that case, the sender will initiate a retransmission. The retransmission must finish within the remaining time in the CAP portion of the current superframe; otherwise, it is deferred to the CAP portion of the next superframe.

The device will repeat the process of transmission until it fails up to a maximum of *macMaxFrameRetries = 3* times. The transmission of an acknowledgment frame in the CAP will commence either *aTurnaroundTime* symbols after the reception of the last symbol of the data or MAC command frame or at the backoff slot boundary.

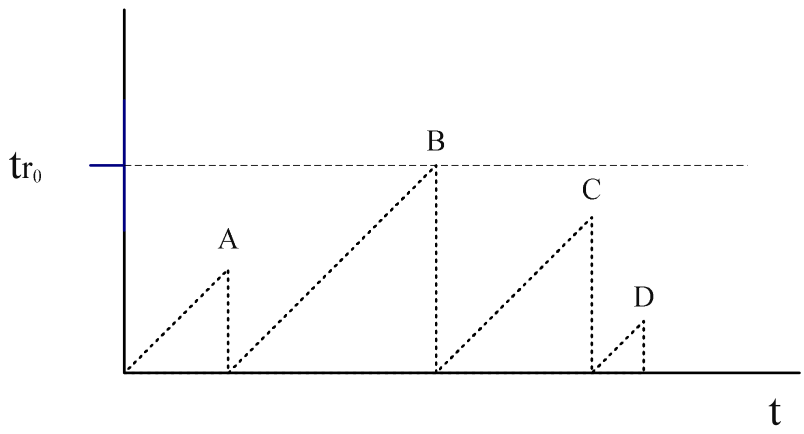


Figure 2.4: A sender continuously sends out packets from $t = 0$. It successfully receives ACK at “A”, “C” and “D”. However the sender needs to retransmit at “B” because the sender does not receive a correct ACK in t_{r_0}.

2.2 DoS and RoQ in WSNs

There are several methods to attack coordinator in WSNs. Denial of Service (*DoS*) attack is one of these schemes [Yu and Xiao, 2006]. However, it is not efficient and can be easily detected. In order to model an attack which is efficient and can avoid the detection as far as possible, Reduction of Quality (*RoQ*) attack is proposed. RoQ attack focuses on optimizing the cost to produce maximum damage and trying to evade the detection simultaneously. RoQ attack is particularly useful in the environment where adversaries have limited energy such as motes in WSNs.

DoS Attacks in WSNs [Ye et al., 2004] [Zhu et al., 2004] have proposed DoS attacks in WSNs that operate by flooding data packets along multi-hop, end-to-end routing paths. Under this attack, along with nodes on the routing paths, the energy will be exhausted quickly. In [Deng et al., 1999], Deng *et al.* presented a similar attack, Path-based DoS (*PDoS*) and proposed a countermeasure to defend against it. The countermeasures make use of One Way Hash Chains (OHC) that is created by one way function periodically. They also described how to deal with the initiation of OHC, the break of OHC and the OHC maintenance.

In [Zhang et al., 2005], Denial-of-Message Attack (*DoM*) and its detection were proposed by McCune *et al.* They considered a scenario that sensor network broadcast protocols are exposed in untrustworthy environment. In this environment, a *DoM* attack will happen when malicious nodes prevent normal nodes from receiving a broadcast message. They modeled and analyzed this attack and present a scheme, SIS (Secure Implicit Sampling), which could detect this attack.

RoQ Attack An effective alternative to DoS and DoM is the recently proposed Reduction of Quality (RoQ) attack schemes [Guirguis et al., 2004b, Guirguis et al., 2004a, Guirguis et al., 2005, Guirguis et al., 2007] which are more potent in terms of the cost/damage tradeoff and are harder-to-detect than traditional DoS attacks. RoQ attacks are particularly useful when adversaries have limited energy resources, such as battery-operated motes in WSNs. Other methods such as the Shrew attack [Kuzmanovic and Knightly, 2006] has similarly exploited the timeout mechanism of TCP in wired networks to bring about a complete denial of service and avoiding detection, but a RoQ attack does not seek to bring about a complete denial of service. The goal is to achieve a high degree of potency. Unfortunately, it is currently difficult to compare RoQ schemes as there are no agreed definitions of the concepts of damage and cost used in various attack models.

2.3 Jamming Attacks

Jamming is a typical attack in wireless networks, which can disrupt wireless communication by emitting interference signals. It is defined as the act of intentionally directing electromagnetic energy toward a communication system to disrupt or prevent signal transmission [Adamy and Adamy, 2004]. IEEE 802.15.4-compliant wireless networks are susceptible to jamming attacks since such networks are composed of small energy-constrained devices to execute some tasks without a central powerful monitoring node.

In physical layer, jamming attacks are implemented by sending out a high transmission power signal that corrupts a communication link or an area. Attackers also can launch jamming attacks at the access layer by either corrupting control packets or occupying the channel for the maximum allowable time, so that the network throughput will decrease because legitimate nodes are not able to access the channel [Negi and Perrig, 2003]. In [Mallik et al., 2000], the authors consider a one-way time-slotted packet radio communication link in the presence of a jammer that works in an on-off mode. There are some other jamming attacks that impact the network layer by injecting a malicious packet along certain routes or on the transport layer, e.g. SYN message flooding. By observing the channel and learning protocols' semantics such as packets sending distribution, slot size, or preamble size, various jamming attacks aimed at different *MAC* protocols in sensor networks are proposed in [Law and Palaniswami, 2009] to attack the network effectively. However, they are either static or not energy-efficient.

Some efficient strategies are proposed in [Lin and Noubir, 2004, Li et al., 2007]. The work in [Lin and Noubir, 2004] discusses a low-energy attack that destroys a packet by jamming only a few bits, such that the code error correction functionality will take on an excessive load. To save energy, a jammer controls the probability of jamming and the transmission range to cause maximal damage to the network in terms of corrupted communication links [Li et al., 2007]. However, the access probability is difficult to get hold of and the transmission range is greatly affected by the surroundings.

In [Xu et al., 2006], in terms of attack strategies, Xu et al. propose four kinds of jammer models: constant, deceptive, random, and reactive jammer. In this paper, jammers refer to attackers or compromised nodes that emit jamming signals. A constant jammer continually emits a radio signal to prevent legitimate nodes from getting hold of the channel. A deceptive jammer constantly injects regular packets to the channel without any gap between packets, so that a normal communicator will be deceived into going into the receive state. A random jammer randomly alternates between sleeping and jamming. This is good for jammers that do not have an unlimited power supply, but the approach offers no guarantees on the effect of the attacks.

Game theory is considered to describe the procedure of jamming and defending. A malicious node corrupts broadcasts from a base station (*BS*) to a sensor network by depriving other nodes from receiving a broadcast message. The procedure of attacking and *BS* defending is formulated as a zero-sum game in [McCune et al., 2005]. A one-way time-slotted packet radio communication link can be attacked by a jammer that works in an on-off mode. This process is modeled as a two-person zero-sum noncooperative dynamic game [Mallik et al., 2000]. However, a more precise model to reflect the dynamic procedure of jamming and defending is Stackelberg game. In Chapter 4, we use Stackelberg game to analyze the behavior of the jammer and the network. Based on the analysis, we proposed a dynamic jamming strategy and finally provide an efficient countermeasure.

2.4 DSSS

Various jamming problems in wireless communication have been widely discussed in the previous works (e.g., [Law and Palaniswami, 2009, Xu et al., 2005, Xu et al., 2006, Negi and Perrig, 2003, Poisel, 2006a]). The common jamming countermeasure is spread spectrum such as FHSS and DSSS [Poisel, 2006b]. In these both technologies, jamming resistance has only possible by using the shared key to generate the sequence of frequencies or spreading codes.

However, pre-share key is not scalable to large systems (e.g. GPS and some military application) and suffers from the network dynamics problem in ad-hoc network. Meanwhile, an attacker may learn the shared key from a compromised or malicious receiver. Recently, some work [Baird et al., 2007, Liu et al., 2010, Strasser and Pöpper, 2008, Pöpper et al., 2009] identify the lack of methods for jamming-resistant communication without the shared key and propose some methods to solve this problem. An uncoordinated frequency hopping (*UFF*) is proposed to reliably establish secret key when there are jammers existing in the wireless network [Strasser and Pöpper, 2008]. Baird et al. present *BBC* algorithm [Baird et al., 2007] which uses concurrent codes in combination with UWB pulse transmissions. A major limitation of *BBC* is the number of pulses that the attacker can insert.

The Randomized Differential DSSS (*RD-DSSS*) [Liu et al., 2010] is presented to guarantee jamming-resistant wireless broadcast communication. To reduce the overhead of communication, authors use an index code to denote a sequence of spread code. However, the index code is vulnerable when the sender transmits bit '1'. Moreover, to resist reactive-jamming, authors permute all codes of the spread message and use a mapping function to reconstruct them. The computational overhead during the reconstructing procedure is large. It cannot meet the requirement of the energy-constrained network in which the overhead is an important performance metric. When design a spread spectrum technology in the energy-constrained network, the balance between the capacity of jamming-resistance and overhead introduced by the technology should be considered as one of the critical design criterion.

2.5 Summary

In this chapter, we first reviewed the existing approaches to launch DoS and RoQ attack in the wireless personal network and discussed the advantages and disadvantages of all of the existing attacks. Next, we focus on jamming attacks, particularly on reactive jamming where the attacker observe the channel first and then use a proper strategy to effectively attack wireless networks. At last, we review the effective countermeasure to resist jamming attacks, spread-spectrum technique. Furthermore, we review the latest technologies that can spread spectrum without the pre-shared key.

Chapter 3

EERoQ: Energy-Efficient Reduction of Quality Attack

In this chapter, we introduce the energy-efficient reduction of quality (*EERoQ*) with its mathematical model. This chapter is organized as follows. In Section 3.1, we briefly introduce *EERoQ* attack. We propose an energy-efficient *RoQ* attack model and give its analysis in Section 3.2. Section 3.3 presents a practical attack strategy and Section 3.4 illustrates its experimental performance in a real wireless sensor network. Finally, we offer our conclusion in Section 3.5.

3.1 Overview

There are several methods to attack coordinators in WSNs. Denial of Service (*DoS*) attack is one of these schemes [Yu and Xiao, 2006]. However, it is not energy-efficient and can be easily detected by statistical mechanism [Xu et al., 2005]. In order to model an attack which is energy-efficient and can avoid the detection as far as possible, Reduction of Quality (*RoQ*) attack is proposed. The *RoQ* attack focuses on optimizing the cost to produce maximum damage and trying to evade the detection simultaneously. *RoQ* attack is particularly useful in the environment where adversaries have limited energy such as motes in WSNs.

The recently proposed Reduction of Quality (*RoQ*) approaches [Guirguis et al., 2004b, Guirguis et al., 2004a, Guirguis et al., 2005, Guirguis et al., 2007] which are more potent in terms of the *cost/damage* tradeoff and are harder-to-detect than traditional *DoS* attacks. *RoQ* attacks are particularly useful when adversaries have limited energy resources, such as battery-operated motes in WSNs. Other methods such as the Shrew attack [Kuzmanovic and Knightly, 2006] has similarly exploited the timeout mechanism of *TCP* in wired networks to bring about a complete denial of service and avoiding detection, but a *RoQ* attack does not seek to bring about a complete denial of service. The goal is to achieve a high degree of potency so that the attacker can make maximum damage to the network of unit energy consumption. In wireless network, the jamming attack is one of the most important techniques to reduce the service quality (e.g. throughput). It is essential to design an energy-efficient jamming attack in energy-constrained networks such as wireless sensor networks [Law and Palaniswami, 2009].

3.2 An Energy-Efficient RoQ Attack Model

In this section, we describe how an adversary can exploit the IEEE 802.15.4 retransmission mechanism to carry out an energy-efficient RoQ attack using burst traffics. First, we present a mathematic model for describing the attack and then describe our method for calculating and optimizing the potency of an attack to design an efficient attack. It means that attack will last longer by means of the optimal potency. Because all of the following analysis is based on burst traffic, we will present a strategy in Section 3.3 to generate the burst traffic. At last we test this strategy practically in Section 3.4.

3.2.1 Mathematical model

The purpose of an energy-efficient RoQ attack is to produce the maximum damage per unit cost. Our strategy seeks to maximize energy efficiency by using burst traffic to block packets, thereby making honest nodes repeatedly enter the retransmission state, greatly decreasing the network throughput. We achieve this by optimizing a metric named potency (in economy, people call it utility). IEEE 802.15.4 uses a simple retransmission mechanism that retries transmission after a fixed interval mAWD when it does not receive or receives the wrong ACK frame. This makes it vulnerable to the periodic generation of bursts of traffic which can force it to retransmit almost every packet. For this purpose, we generate a periodically repeating short-duration burst having a time-scale length of t_2 (Figure 2.1(b)).

There are two main factors of the success of our RoQ attack, when and for how long to generate the burst traffic. A burst is less detectable but it must still be long enough to block packets. As shown in Figure 3.1, a triple (l, T, P) can describe our RoQ attack model. l denotes the duration of burst traffic blocking frame transmission. T denotes the attack cycle and P is the packet drop rate of device during the burst traffic l. In order to attack effectively, the RoQ attack burst traffic length should last long enough to block data or ACK frames. As shown in Figure 2.1, the sender needs $t_1 + t_2 + t_3$ to receive ACK and t_1 is equal to t_3 which approaches zero. Hence, l should be bigger than t_2. Simultaneously, the RoQ attack is to avoid detection by sending a low average volume of traffic. Thus, the value of l should be much smaller than T. For example, we can set $l < \beta \cdot T$ and β is a value smaller than 0.5. And T's value should be limited by a boundary to promise the effect of attack.

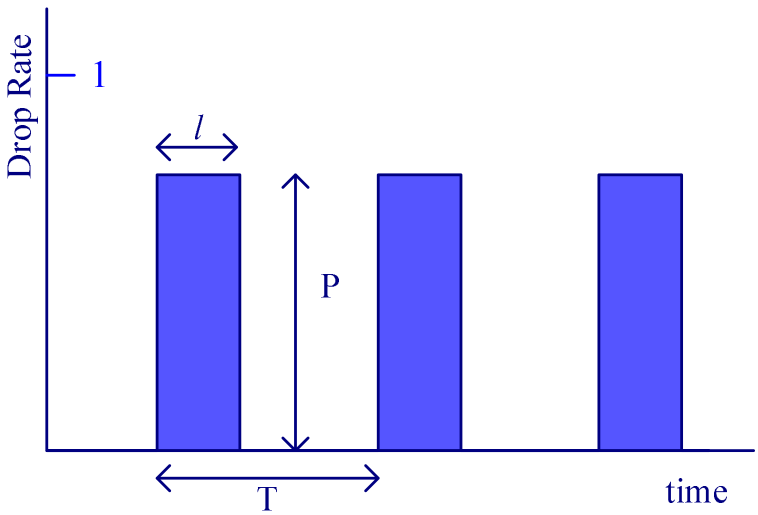


Figure 3.1: Attack Model.

3.2.2 Calculating Potency

In this section, we use a mathematical model to calculate the damage and cost, and therewith the potency, of an attack. Potency is defined as the ratio of damage to the cost of launching an attack in term of energy consumption. It reveals how the effect of attack changes to reflect the energy consumption.

$$Potency = \frac{Damage}{Cost} \tag{3.2.1}$$

To compute damage, we calculate the reduction of network normalized throughput under the *EERoQ* attack. We initially also considered trying to take into account the damage of the additional energy consumption of a network caused by an attack, but found this at this stage too difficult to measure.

One way to deduce throughput is to use a nondimensionalized scheme such as normalization, where normalized throughput is the ratio of the available bandwidth of devices transferring data when under attack and not under attack.

To calculate normal and abnormal throughput, we denote the length of a burst of traffic as l, the drop rate as P and the attack period as T. The drop rate is the ratio of the number of packets received during an attack against the number received during normal operations. The maximum wait for an acknowledgment frame to arrive following a transmitted data frame is denoted as *macAckWaitDuration (mAWD)*.

To show how this calculation might work, let's first consider an ideal situation wherein attackers can generate perfect burst traffic during which all packets will be blocked. The normalized throughput is the ratio of the expectation value of available bandwidth (EB) to total bandwidth.

$$\rho = \frac{\lceil \frac{mAWD}{T} \rceil T - mAWD}{\lceil \frac{mAWD}{T} \rceil T} \tag{3.2.2}$$

In a non-ideal situation, it is not the case that all of the packets will be blocked in the burst traffic period. We use a parameter named drop rate P to describe the packet drop rate during burst traffic. The normalized throughput is the ratio of the expectation value of available bandwidth (EB) to total bandwidth.

$$\rho = \frac{EB}{\lceil \frac{mAWD}{T} \rceil T} \tag{3.2.3}$$

There are two situations in which the packet drop rate of a sender is smaller than one. 1) During burst traffic period, the burst traffic may block a packet or nothing. During the burst traffic period, the probability of blocking a packet is P and the available bandwidth is

$$\lceil \frac{mAWD}{T} \rceil T - mAWD$$

2) the probability of all packet escaping being blocked is $1-P$ and the available bandwidth is

$$\lceil \frac{mAWD}{T} \rceil T$$

.

Thus the expectation value of available bandwidth is

$$EB = P(\lceil \frac{mAWD}{T} \rceil T - mAWD) + (1 - P)\lceil \frac{mAWD}{T} \rceil T \tag{3.2.4}$$

The normalized throughput will increase up to:

$$\rho = P\frac{\lceil \frac{mAWD}{T} \rceil T - mAWD}{\lceil \frac{mAWD}{T} \rceil T} + (1 - P) \tag{3.2.5}$$

Damage is the reduction of normalized throughput under our *RoQ* attack. The original value of normalized throughput when there has not been an attack is 1 and we assume that the normalized throughput under *RoQ* attack is ρ Therefore, we use $1 - \rho$ to depict the network capacity that is depleted by attacks. Damage is calculated as follows:

$$\begin{aligned} D &= 1 - \rho \\ &= 1 - (P\frac{\lceil \frac{mAWD}{T} \rceil T - mAWD}{\lceil \frac{mAWD}{T} \rceil T} + (1 - P)) \\ &= P - P\frac{\lceil \frac{mAWD}{T} \rceil T - mAWD}{\lceil \frac{mAWD}{T} \rceil T} \\ &= P \cdot \frac{mAWD}{\lceil \frac{mAWD}{T} \rceil T} \end{aligned} \tag{3.2.6}$$

The calculation of potency requires us to deduce the metric of cost. We propose two different metrics to do this: time domain cost (CT) and relative cost (CR).

Time-Domain Cost (C_T): We can calculate the cost to the attacker of this attack using the ratio of the wake-up period to the cycle. In our attack model, adversaries generate a burst traffic, which lasts for time l, periodically (T) and $l < T$. In the interval $(T - l)$, attackers can sleep and so have an energy consumption near to zero. We assume that the average consumption of energy in each l is equal. Obviously then, a smaller l, means less energy is consumed. The time-domain cost, CT (it is often called duty-cycle.) is given by:

$$C_T = \frac{l}{T} \tag{3.2.7}$$

Relative Cost (C_R): To calculate the relative cost of an attack, we must determine how much power is consumed when an attack is under way (when nodes are compromised and are in fact attack nodes) and when nodes are benign or ”honest”. In this case, the relative cost, CR is given by

$$C_R = \frac{cost\_of\_attackers}{cost\_of\_honest\_node} \tag{3.2.8}$$

In this chapter, we use the time-domain cost. We can compute the attack potency defined in Eq. 3.2.1 by using the equations for damage (4.3.2) and cost (3.2.7) as follows.

$$\begin{aligned} Potency &= \frac{D}{C_T} \\ &= \frac{P \cdot \frac{mAWD}{\lceil \frac{mAWD}{T} \rceil T}}{\frac{l}{T}} \\ &= \frac{P}{l} \cdot \frac{mAWD}{\lceil \frac{mAWD}{T} \rceil} \end{aligned} \tag{3.2.9}$$

We get the following relation from Eq. 3.2.9.

$$Potency \leq P \cdot mAWD/l$$

This inequality can be derived from the fact that $\lceil mAWD/T \rceil \geq 1$. Thus the upper bound of potency is $P \cdot mAWD/l$.

The purpose of introducing potency and optimizing it is to balance consumption of an attacker against damage. We must promise the effect of attack when we try to lower the cost. We set a threshold (θ) and damage should be greater than (θ). In Eq. 4.3.2 we find $\theta \leq D \leq 1$ and $0 < \theta \leq 1$. To solve this inequality, we divide the value of θ into three cases: $(0, P/2]$, $(P/2, p]$ and $(P, 1]$. In these three cases, we get the following T.

$$\begin{cases} T \geq T_1, & 0 < \theta \leq P/2. \\ \frac{1}{2} \leq T < T_2, & 1 < T < T_3 \text{ when } p/2 < \theta \leq P; \\ T \in \phi, & P < \theta \leq 1. \end{cases} \tag{3.2.10}$$

The value T_1, T_2 and T_3 satisfy the following equation and ranges:

$$P \cdot \frac{mAWD}{T} \cdot \lceil \frac{mAWD}{T} \rceil^{-1} = \theta$$

$$\begin{cases} T_1 \geq 2 \cdot mAWD \\ mAWD \leq T_2 < 2 \cdot mAWD \\ mAWD/2 \leq T_3 < mAWD \end{cases}$$

This tells us that the best interval between burst traffics is T_1 or T_3 and that the range of θ for each interval is respectively $(0, P/2]$ and $(P/2, p]$.

From the above optimal result, we know that a proper setting of T (Eq. 3.2.10) will lead to an efficient attack. For example, our attack must be more efficient than the *continuous jamming*. In continuous jamming, l is equal to T. Based on Eq. 3.2.9, the continuous jamming potency will not larger than our attack. It can derive from the fact $l \leq T$ in the *RoQ* attack. It means that our attack scheme will last for longer time than the continuous style under the same attack damage. We also can prove that our attack is more potent than the *random jamming*.

This RoQ attack model is based on the generation of burst traffic. We will present a strategy in Section 3.3 to realize it in the network composed of MicaZ motes. The generated burst traffic does not target at blocking packets completely. We test the strategy practically in Section 3.4 to set parameters to maximize the RoQ effect.

3.3 Generating Burst Traffic

In this section, we propose a strategy for generating burst traffic that can force devices to enter the retransmission state in IEEE 802.15.4 compatible wireless sensor networks. This scheme exploits the contention-based channel access mechanism CSMA-CA. Other schemes do exist for forcing dropped packets [Li et al., 2007, Law and Palaniswami, 2009, Cagalj et al., 2005, Xu et al., 2005, Xu et al., 2006, Sun et al., 2007, Karlof and Wagner, 2003, Kyasanur and Vaidya, 2005] but our focus is on CAP in IEEE 802.15.4.

Our goal is that attackers should get access to the channel more efficiently and frequently than honest nodes and the approach is to exploit the channel contention mechanism CSMA-CA which is vulnerable to shortening or ridding the backoff period and avoiding the assessment of channel.

One simple, but inefficient, way to degrade the network performance of a network using CSMA-CA is simply to flood the network by transmitting a large number of packets in a short time. This can be done by using a completely functional IEEE 802.15.4 device that strictly conforms to CSMA-CA. Such a high packet arrival rate, however, is easy to detect using statistical mechanisms and has a considerable energy cost. A less expensive approach is to exploit CSMA-CA in the MAC layer of IEEE 802.15.4, where it is employed as the contention-based medium access mechanism and is available to be modified or even ignored.

CSMA-CA makes use of three variables to ensure the fairness of the channel contention process, a process in which devices compete for access to the transmission channel. These three variables are clear channel assessment (CCA), contention window (CW), and backoff exponent (BE). We assume that clear channel assessment is working normally when CCA is set to "W". When CCA is equal to 1, the PHY layer tells the MAC layer that the channel is busy all the time without any channel detection. If CCA is equal to 0, the PHY layer tell the MAC layer that the channel is idle. When CW is equal to k, a device should detect that the channel is idle k times continuously before sending out a packet. The default value of CW is 2 in CSMA-CA. Devices which want to send a packet must delay for a random period of time ($random(2^{BE} - 1)$) before detecting the channel and finish the following procedure. If BE is equal to "W", it means the value of BE is initiated and increase as specified in IEEE 802.15.4. When BE is equal to j, it means that BE will be initiated as j and does not increase in the following procedure.

When CW is equal to k, it means that a device should detect that the channel is idle k times continuously before send out a packet. The default value of CW is 2 in CSMA-CA. Devices which want to send a packet must delay for a random period of time ($random(2^{BE}-1)$) before detect the channel and finish the following procedure. If BE is equal to "W", it means the value of BE is initiated and increase as specified in IEEE 802.15.4. When BE is equal to j, it means that BE will be initiated as j and does not increase in the following procedure. These three variables join together to ensure the fairness of channel contention. However, for attackers, they will be modified to increase the possibility to occupy the channel.

This process can be quite simply interfered with so as to ensure that channel contention is not fair and in fact favors an attacker. This is done in three ways. One way is to set a smaller initial backoff value without increasing the backoff exponent, BE, after a collision. This shortens the random backoff countdown, reducing the number of required CCA attempts (CW) from two to one and thereby giving the attacker twice as many opportunities to occupy the channel. Another way is to omit the regular CCA function, which will allow the attacker to start transmitting immediately after finishing the random backoff countdown. Third, an adversary can avoid the random backoff countdown by setting BE to 0, causing the honest node, which cannot access the channel at all, to increase its own backoff exponent, making it even more difficult for itself to access the channel.

Although adversaries can send out a packet at will, these modifications are useful to reduce the active period of attackers and send out malicious packets whenever attackers wish but not used to send out malicious packets arbitrary. Omitting CCA, setting CW to 1 and setting BE to 0 can reduce waiting time approximately to 0 before send out malicious packets.

The attacker should not send out malicious packages randomly or continuously because we assume that attackers are also limited-power wireless sensor nodes. The random scheme would waste the energy and you can not estimate the power consumption. The continuous one would make the attack die soon because of the continuous enormous power consumption. Attackers would balance their power consumption and the damage to the network they belong.

The modified CSMA-CA parameters can be set as below and the strategy using the fourth setting is illustrated in Algorithm. 1.

- Setting 1: (BE, CCA, CW) = (W, W, 2)
- Setting 2: (BE, CCA, CW) = (0, W, 2)
- Setting 3: (BE, CCA, CW) = (0, W, 1)
- Setting 4: (BE, CCA, CW) = (0, 0, 1)

The best settings for the RoQ attack are illustrated as in the fourth setting, with (BE, CCA, CW) as respectively (0, 0, 1) as these settings produce the shortest waiting time. BE set to 0 means the attacker does not need to wait before assessing the channel status. CCA set to 0 means the attacker assess the channel as always idle. CW set to 1 means we do not need to this assessment once.

Although attackers can consume the energy of coordinator by requiring acknowledgement, it will make the modification of CSMA-CA meaningless because attackers have to spend more time on waiting for his corresponding ACK frame. We focus on how to reduce the throughput of network. Attackers should achieve the channel more frequently than honest nodes or just send out packets which work as interference signals.

3.4 Experiments

In this section, we describe our experiments with each strategy describe in Section 3.3 to establish the parameters that would maximize the RoQ effect. We conducted our experiments on an IEEE 802.15.4-compliant sensor cluster implemented by [Severino et al.,], using MicaZ with TinyOS as the platform. The cluster operates in the ISM band at 2.4 GHz. The target node is the coordinator and we degrade the throughput of the networks by producing periodic short bursts of traffic and exploiting the vulnerability of the retransmission mechanism. The fact that the coordinator must receive the packet from its neighbors by one-hop no matter what path a packet took to get there, means that we can conduct our experiment with one coordinator and several devices as its neighbors (the circle shown in Figure 2.1(a)) to simulate the data aggregation in a network. Table 3.1 lists the abbreviations used in this section.

3.4.1 Configuration

In each of the following four experiments, devices send packets with 2-byte payloads in the fixed frequency to the coordinator. The payloads record sequence numbers of the packets, count the number of packets, and r analyzing the extent of damage. In each experiment, we adjust the attack parameters, length of packet,

Table 3.1: Abbreviation.

Symbol	Meaning
DR	$DR = (N_s - N_r)/N_s.N_r$ is the number of packets arriving at the coordinator in the absence of attacks and N_s is the number of packets arriving at the coordinator in the presence of attacks.
NA	Number of Attacker(s)
BE	Backoff Exponent
CW	contention Window
CCA	Clear Channel Assessment
APL	Packet Payload Length of Attacker Packet Sending Frequency of Attacker
ASF	Packet Sending Frequency of Attacker

and the settings of BE, CW and CCA in order to test which scheme or combination of schemes is best, We used a PC to communicate with the coordinator and to count the packets sent to it from devices. We used the receipt mechanism of IEEE 802.15.4 to insert a function to send the received packet to the PC when the physical layer of coordinator receives a packet. In IEEE 802.15.4, when the physical layer receives a packet, it activates an event function called MCPS (MAC Common Part Sublayer) DATA.indication to notify the MAC layer that a packet was received by the PHY layer. We insert a function which will forward received packets to the PC. To identify and count the packet, we give a unique ID to each node that identifies the device and attacker. This ID is placed at the head of packets. When the packets are forwarded to the PC, they are written to files. The PC runs a C program to maintain files and count the number of type packets. The program will filter the wrong packets. They will not be counted.

IEEE 802.15.4 runs on TinyOS in our experiments. We can use the mote-PC serial communication functions provided by TinyOS to forward the packets to the PC. These functions require the setting of appropriate variables, for example, PORT and SPEED to receive the forwarded packet properly.

3.4.2 Experiment Results

To determine the best setting, we did four experiments on an IEEE 802.15.4 compatible wireless sensor network, varying the number of packets, sending packets frequency, backoff delay, contention window and CCA of attackers to observe the packet drop rate of the network.

In the first experiment 3.2, attackers do not modify any portion of the CAMA-CA protocol. It means that attackers need double ($CW = 2$) idle channel report before sending out malicious packets. Prior to detecting the channel status, they have to wait for $2^{BE} - 1$ symbol and BE follows the additive increase principle. At the same time, the detection result from CCA is veritable. This means that the PHY layer executes the detection program exactly and returns the correct result. Figure 3.2 shows that as ASF and APL increase, so does DR.

In the second experiment 3.3, we set BE to 0 in the CSMA-CA mechanism running on malicious nodes. Attackers must wait for the $random(2^{BE} - 1) =$

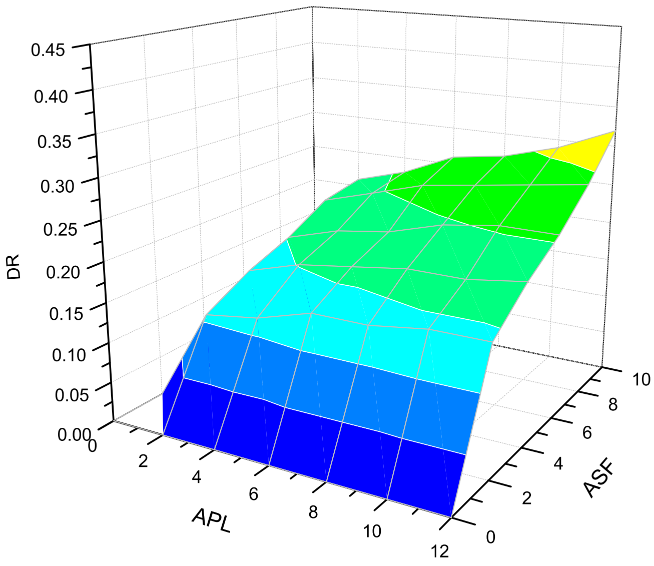


Figure 3.2: Experiment 1: Attacker follows CSMA-CA.

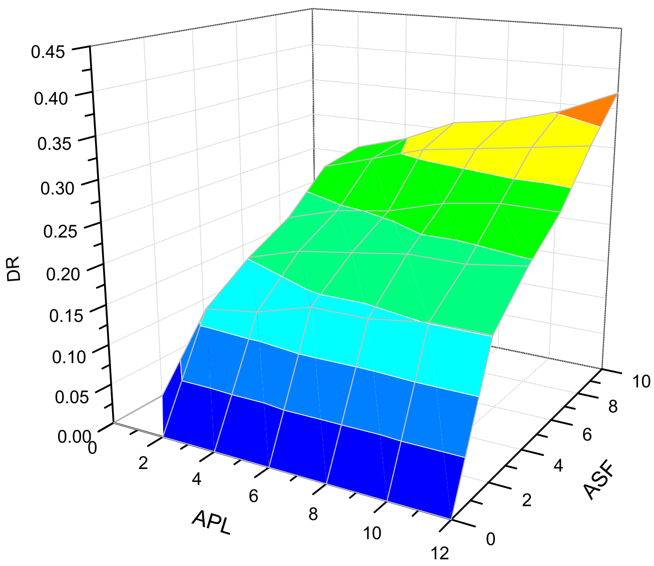


Figure 3.3: Experiment 2: Modify BE.

0 symbol before detecting the channel status and BE will not increase after a failure to send out packets. This means that when attackers start the CSMA-CA mechanism, they access the channel immediately using CCA. The reduction of latency time implies that the idle state of attackers is shortened. Hence the power consumption will decrease. Figure 3.3 shows that this setting does not decrease the drop rate even if it is increased by small amounts. This setting can thus simultaneously maintain or increase the potency of attack in that it increases damage with low power consumption.

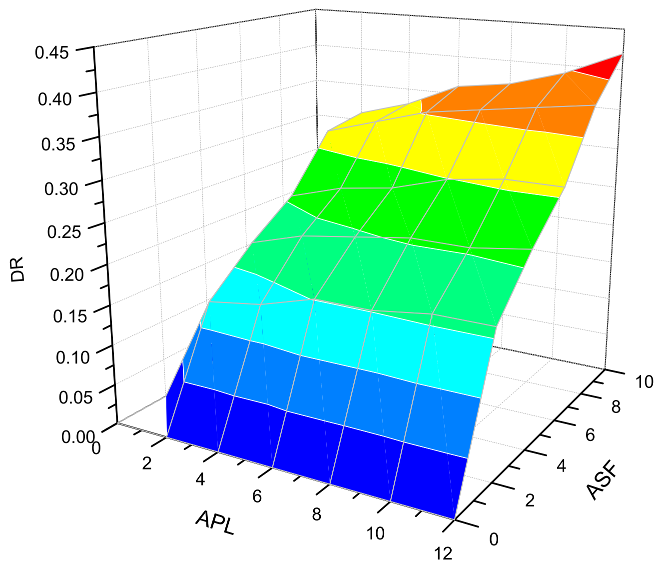


Figure 3.4: Experiment 3: Modify BE and CW.

In the third experiment 3.4, we decrease CW to 1 and maintain the other parameters as in the second experiment. This means that attackers will send out malicious packets immediately that they detect the channel is clear after a backoff period. Again as in the second experiment, attackers do not need to wait before detecting the channel. This setting means that when malicious nodes enter the CSMA-CA mechanism, the malicious nodes will send out packets after one clear channel detection. Reducing the number of CCA also can reduce the power consumption because the PHY layer can reduce execution and malicious nodes do not need to wait for another channel detection. To remove the possibility of a second assessment that might return a busy signal, we set CW to 1. These setting increase the probability of gaining access to the channel and reduce power consumption.

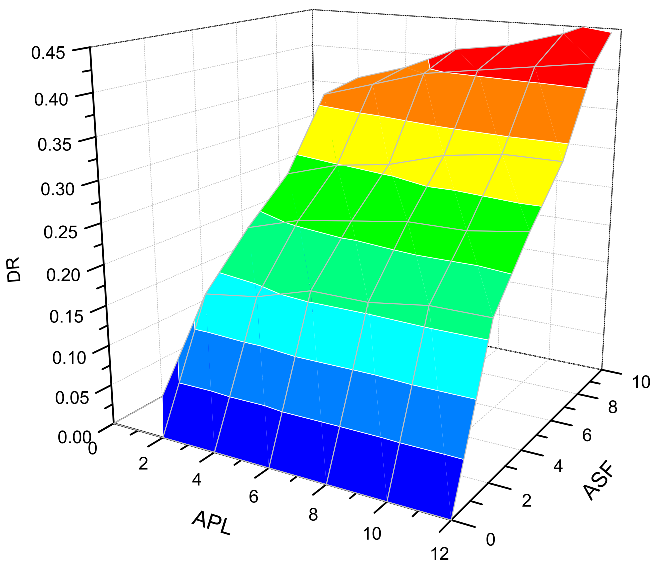


Figure 3.5: Experiment 4: Modify BE, CW and CCA.

In the fourth experiment 3.5, we do not let CCA work correctly even close it to save power and it will always return “IDLE” without any detection. We also set CW to 1 and BE to 0 to reduce the latency time. CW will fall to 0 immediately because the modification of CCA. As expected, this setting is the best because it seldom requires a second wait to send out the packets and malicious nodes will have more opportunities to occupy the channel.

We can conclude that the packet drop rate will increase and energy consumption will decrease when attackers simultaneously maintain BE as a constant (e.g. 0 or 1) and turn off the CCA function. The procedure of efficient modified CSMA-CA for attackers has been shown in Algorithm 1. These experiments show that using motes to generate burst traffic is feasible and the packet drop rate (P) of senders during the burst traffic period will greatly affect the potency of attackers.

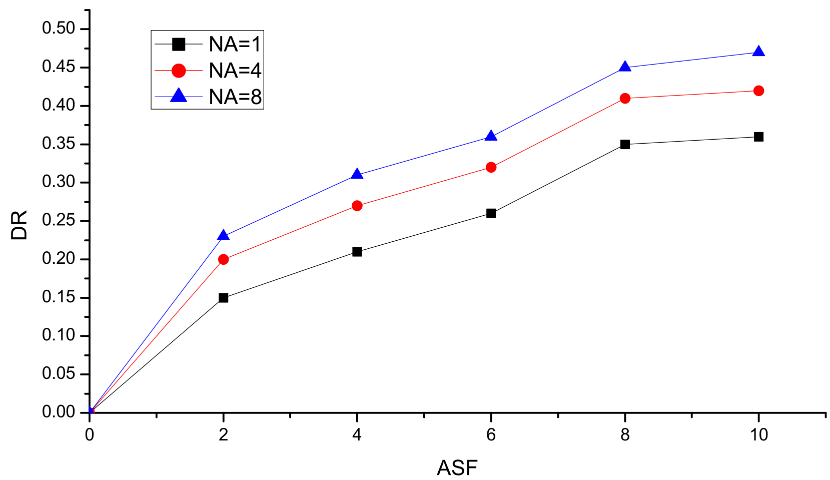


Figure 3.6: Drop rate versus packet sending frequency of attacker when APL is 2.

Our experimental results also show that the packet drop rate to senders will be the highest when we configure (BE, CCA, CW) = (0, 0, 1). We use this configuration to evaluate how NA and ASF affect DR. By taking $APL = 2, 4, 8$ and 12, we got four groups of three curves with 1, 4 and 8 attackers as shown in Figure 3.6,

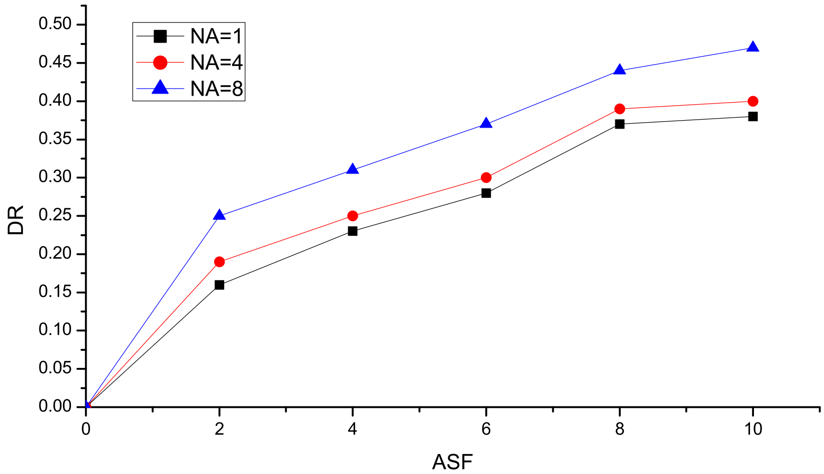


Figure 3.7: Drop rate versus packet sending frequency of attacker when APL is 4.

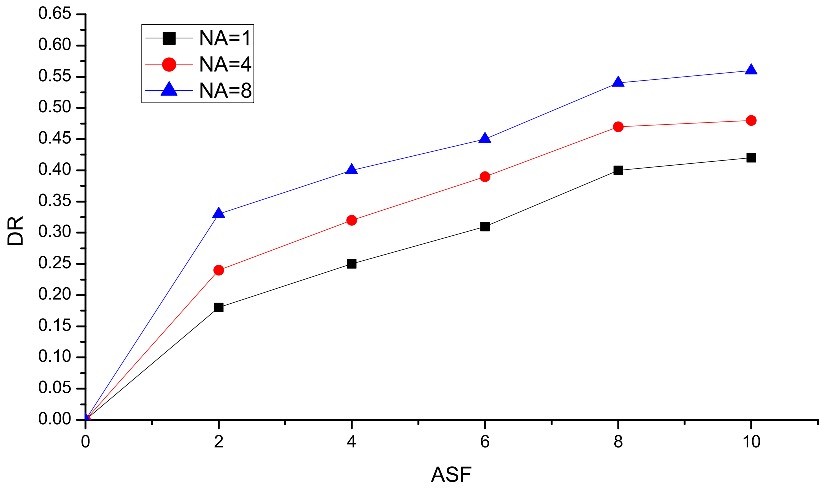


Figure 3.8: Drop rate versus packet sending frequency of attacker when APL is 8.

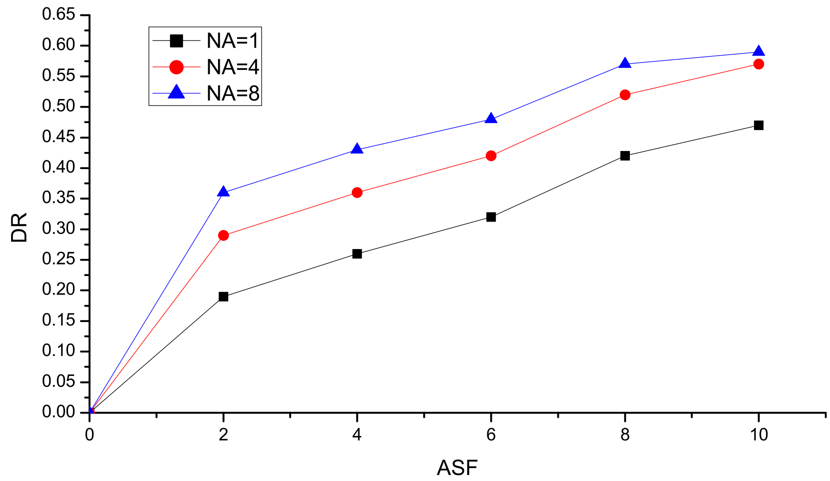


Figure 3.9: Drop rate versus packet sending frequency of attacker when APL is 12.

3.7, 3.8 and Figure 3.9. This indicates that increasing NA and ASF will increase the packet drop rate. Using a higher packet sending frequency is better than using more attackers, because their contributions to the drop rate are similar but increasing the sending frequency is easier than occupying more motes to launch the RoQ attack.

3.5 Summary

In this chapter, we have proposed an energy-efficient RoQ attack against wireless sensor networks based on IEEE 802.15.4. We use a metric called potency to describe the cost/damage ratio of an attack and formulate a mathematical model to optimize potency. We present an efficient attack based on the optimal potency. This model allowed us to determine a feasible strategy for generating burst traffic by exploiting the CSMA-CA mechanism. Moreover, we propose a burst traffic strategy which is energy-efficient in the sense that an attacker can reduce the back-off period to 0 and close the CCA function to save power. We conducted experiments, using a network that is composed of real sensor motes, to implement the strategy of generating burst traffic to force the sender to enter the retransmission state. The experimental results show that an attacker can modify the CSMA-CA mechanism to increase its ability to access the wireless channel and to further reduce the network throughput.

Algorithm 1 A strategy using the best setting.

Initialization: $NB = 1$, $CW = 1$, $BE = 0$

for $NB <= macMaxCSMABackoffs$ **do**

  No backoff period because random$(2^{BE} - 1) = 0$

  for $CW > 0$ **do**

    if Channel is idle **then**

      channel will be always idle because of the malicious implanted function.

      $CW - -$;

    else

      break;

    end if

    if $CW == 0$ **then**

      return SUCCESS;

    end if

  end for

  $CW = 2, NB + +, BE = min(BE + 1, macMaxBE)$

end for

return Failure;

Chapter 4

A Stackelberg Game for Dynamic Jamming Attack and Defense

In this chapter, we investigate the problem of dynamic jamming attacks which is smarter than the fixed-period jamming attack proposed in Chapter 3. Most existing works on jamming attacks fall into one of the following two categories according to whether the network configuration is known by attackers. In the first category [Xu et al., 2006, Law and Palaniswami, 2009], jammers are unaware of the network configuration, e.g. the routing protocol, network topology, or retransmission mechanism. The second category assumes that jammers are aware of the network configuration so that a jammer can adopt a relevant strategy of attack. However they are not energy-efficient methods of attack because the jammer consumes energy sooner than the victims, given comparable energy budgets. In this chapter, we propose an effective dynamic jamming attack (*EDJam*) that is feasible to launch and difficult to defend. The jammer adjusts the period of its jamming attack in order to achieve maximal attack utility, with more damage done to network at less cost to launch the jam. Likewise, as a defender, the network would dynamically select a retransmission mechanism to defend against *EDJam* to maximize its utility of high throughput and reliability. In order for the jammer to maximize its utility, it needs to know the current value of the network retransmission timer (the longest waiting time for the ACK frame). Accordingly, the network would need to know the current period of jamming. Therefore, we use a dynamic competition model to describe the procedure of attackers jamming and the network defending.

This chapter is organized as follows. In Section 4.1, we briefly introduce the work. Section 4.2 gives a detailed description of the network and attack model, which focuses on the exploitation of the vulnerability of the 802.15.4 retransmission mechanism. Section 4.3 uses the Stackelberg game to formulate the model and presents the equilibrium analysis. Completing the analysis of the performance of the model, we simulate our model in Section 4.4. Section 4.5 contains the conclusion.

4.1 Overview

Game theory is currently considered to describe the procedure of jamming and defending. A malicious node corrupts broadcasts from a base station (*BS*) to a sensor network by depriving other nodes from receiving a broadcast message. The procedure of attacking and *BS* defending is formulated as a zero-sum game in [McCune et al., 2005]. A one-way time-slotted packet radio communication link can be attacked by a jammer that works in an on-off mode. This process is modeled as a a two-person zero-sum noncooperative dynamic game [Mallik et al., 2000]. However, a more precise model to reflect the dynamic procedure of jamming and defending is Stackelberg game. In this chapter, we use Stackelberg game to analyze the behavior of the jammer and the network. Based on the analysis, we proposed a dynamic jamming strategy and finally provide an efficient countermeasure.

4.2 Attack Model

In this section, we first simply review the retransmission mechanism used in 802.15.4 and demonstrate how an attacker can exploit this retransmission mechanism to launch an *EDJam* attack. Then, we present a model to describe the attack.

In the open wireless circumstance, there are many factors that would lead to the loss of data. Therefore, a retransmission mechanism is necessary for *WPANs*. In the retransmission-enable 802.15.4, if the destination receives the frame correctly, it will generate and send back an acknowledgement. In this chapter, device and sender, coordinator and receiver, are exchangeable. *macAckWaitDuration* is the maximum number of symbols (default value is 16 *microseconds*.) to wait for an acknowledgment frame to arrive following a transmitted data frame. In this chapter, we denote the value of *macAckWaitDuration* as t_{r_0} and name it retransmission timer. If the expected acknowledgement is received within t_{r_0} symbols after the original data frame, the sender will believe that the transmission is successful. If the sender does not receive the acknowledgement within t_{r_0} symbols or receive a wrong acknowledgement, the sender will conclude that the transmission has failed. In that case, the sender will initiate a retransmission.

This retransmission mechanism, while essential for robust communication, provides an opportunity for our *EDJam* attack. The purpose of an *EDJam* attack is to force the sender to continuously enter the retransmission state by jamming signals with dynamic period. The jamming period is decided by observing the value of the network current retransmission timer. A jammer can calculate the value of the retransmission timer by blocking a packet sent by a legitimate node in the network. First, using a sniffer, a jammer detects a sender sending out a packet at time x and block the packet or its ACK frame by jamming signals. Then, the sender will retransmit the same packet again and the jammer would detect it at time y. Therefore, the value of the current retransmission timer is y-x.

Our attack model is visualized in Figure 4.1, in which we use a triple *(l, T, P)* for description. l denotes the duration of the jamming signal for blocking the frame transmission. T denotes the attack period, which means the interval between two continuous jamming signals. P is the packet drop rate during the jamming signal. We use the parameter P to measure the jamming signal strength. P is defined as the ratio of the number of packets that have arrived to the number of packets that were sent out. The jammer will adjust its jamming period (T_i, T_j)

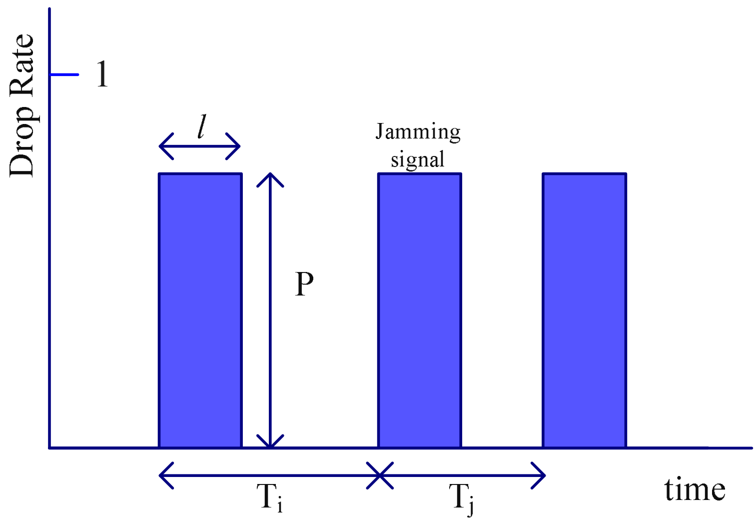


Figure 4.1: In the attack model, l is the duration of the jamming signal. T_i and T_j are the optimal jamming periods at different times under an *EDJam* attack. P is the packet drop rate during the jamming signals, which changes with the strength of the signals.

according the current observation of the network.

There are two main factors in a successful launch of an *EDJam* attack: the duration of each jamming (l) and the jamming period (T). Although a short jamming signal is less detectable, it must still be long enough to effectively block packets. For this purpose, we generate a periodically repeating short jamming having a time-scale length of t_2. Simultaneously, the *EDJam* attack is to avoid detection by sending a low average volume of traffic. Thus, the value of l should be smaller than T. T should be bigger than t_{r_0} so that the jammer can work as a legitimate member in the network. T is a critical parameter to launch our attack efficiently. In Section 4.3, we will derive its optimal value to produce maximum utility by using game theory.

4.3 Game Between Jammers and the Network

In this section, we first define utility functions for both the attacker and the network to formulate the attacking and defending procedure as a Stackelberg game. Then, we study the existence of the equilibrium of the game by maximizing utilities and propose a dynamic network defense strategy. Finally, we prove the uniqueness of the equilibrium in a theorem.

The Stackelberg game can be used to formulate a dynamic process of competition between two players, which can precisely reflect the procedure of the dynamic jamming attack and network defense. We assume that both the attacker and the legitimate nodes (device or coordinator) in the network are rational and selfish. Figure 4.2 shows the procedure of dynamic jamming and network defending. The attacker and network choose their strategy after they observe a competitor's operation. π_a and π_d are the utilities of the attackers and the network respectively. A device uses the standard IEEE 802.15.4 and presents in the network first (Initiated Network). Therefore, it has the right to decide its retransmission mechanism by changing the value of the retransmission timer, so as to maximize its own utility in terms of both performance and reliability. We use t_r to denote the value of the current modified retransmission timer Hence, the network can modify the retransmission mechanism by changing t_r. Attackers in the network choose its jamming period (T), duration (l) and packet drop rate (P) in the predefined network to optimize its utility in terms of the reduction of throughput minus the cost of jamming the network. Therefore, it is a typical two-stage leader-follower game which can be analyzed under the Stackelberg game framework. As shown in Figure 4.2, device, the leader of the game, optimizes its strategy based on the knowledge of the effects of its decision on the behavior of the followers (attackers).

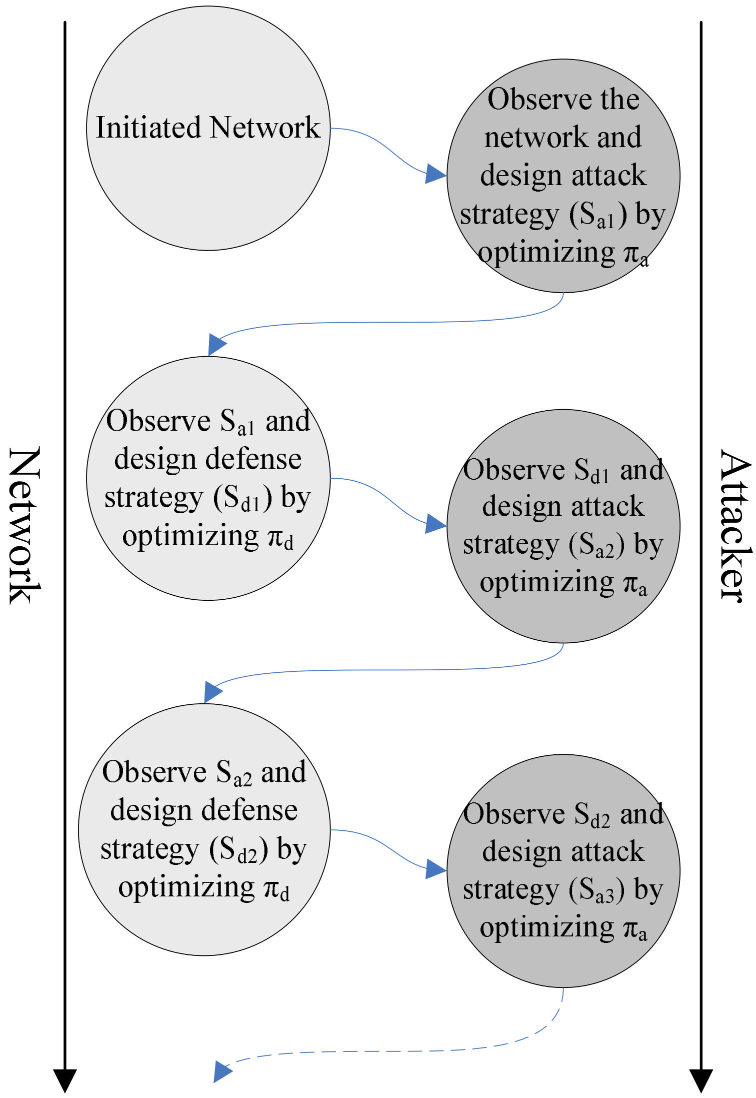


Figure 4.2: The procedure of the Stackelberg game for a dynamic jamming attack and network defense.

4.3.1 Utility Functions

We define the attacker's utility (π_a) as the revenue (damage to the network) minus the cost it incurred in making the attack. The utility of the network is defined to be the performance plus reliability. Performance is measured by the normalized throughput and reliability is measured by the satisfaction function of the retransmission timer. Based on the model illustrated in Section 4.2, we calculate these two utilities in the following manner.

The utility of attacker is:

$$\pi_a = Damage - Cost$$

where *Damage* is defined to be the reduction in network normalized throughput, which is used to measure the performance of the network, under an *EDJam* attack. $Cost$ (Eq. 4.3.1) is proportional to l and P, but in inverse proportion to T.

$$Cost = \frac{l \cdot P}{T} \tag{4.3.1}$$

We derive the normalized throughput on the time domain. In this chapter, the normalized throughput is defined to be the ratio of the channel available time to total time.

$$\rho = \rho(t_r, T) = P\frac{\lceil \frac{t_r}{T} \rceil T - t_r}{\lceil \frac{t_r}{T} \rceil T} + (1 - P) = 1 - P\frac{t_r}{\lceil \frac{t_r}{T} \rceil T}$$

"Damage" is defined as the reduction of network normalized throughput under the attack. When there are no attacks, the value of the normalized throughput is 1. The value under our attack is denoted as ρ. Therefore, we use $1-\rho$ to represent the network capacity that is depleted by the attack. Damage is calculated as follows:

$$\begin{aligned} Damage &= 1 - \rho \\ &= 1 - P\frac{\lceil \frac{t_r}{T} \rceil T - t_r}{\lceil \frac{t_r}{T} \rceil T} \end{aligned} \tag{4.3.2}$$

Therefore, the utility of the attackers is:

$$\begin{aligned} \pi_a &= (1 - \rho) - \frac{l \cdot P}{T} \\ &= \frac{P}{T}(\frac{t_r}{\lceil \frac{t_r}{T} \rceil} - l) \end{aligned} \tag{4.3.3}$$

Now we turn to deriving the network utility, which is defined as normalized throughput plus the network satisfaction index of current modified retransmission mechanism. Although changing the retransmission mechanism could help the network defend against the attack and raise the throughput under the attack, there are some side effects to this approach. Therefore, when we change the default value of

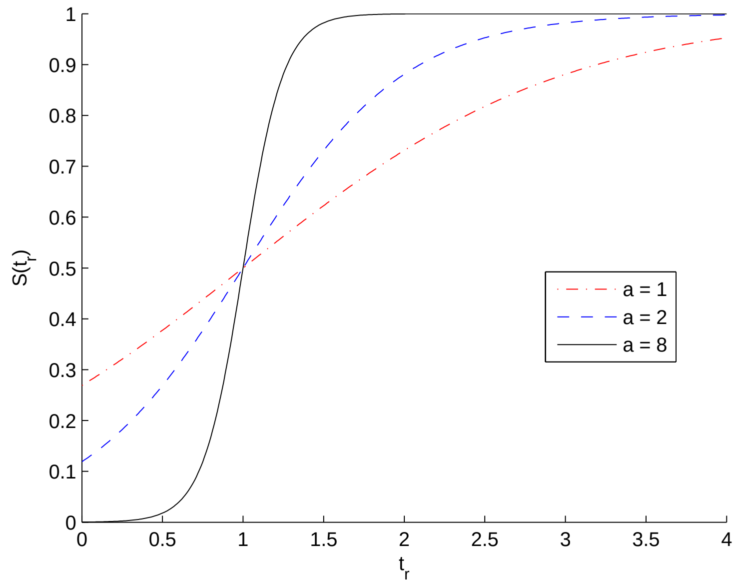


Figure 4.3: Network satisfactory function of the retransmission mechanism.

the retransmission timer, the performance of the retransmission mechanism will change at the same time, and we have to consider how this change impacts the utility of devices. In order to keep the same metric with normalized throughput, we normalize the factor, which is measured by a satisfactory function (Eq. 4.3.4). Sigmoid function [Seggern, 2006] has been widely used to approximate the users' satisfaction with respect to service qualities. a is the steepness of the satisfaction curve. As shown in (Figure 4.3), the meaning of this function is that when the value of the initial retransmission timer becomes larger, the device will be more satisfied with the retransmission mechanism because more reliable communication is promised. And we know that when t_r is larger than a constant, such as the default value suggested in IEEE 802.15.4, the satisfaction of device will increase more and more slowly.

$$S(t_r) = \frac{1}{1 + e^{-a(t_r - t_{r_0})}} \tag{4.3.4}$$

Therefore, the utility of devices is:

$$\begin{aligned} \pi_d &= \rho + S(t_r) \\ &= 1 - P\frac{t_r}{\lceil \frac{t_r}{T} \rceil T} + \frac{1}{1 + e^{-a(t_r - t_{r_0})}} \end{aligned} \tag{4.3.5}$$

4.3.2 Maximizing the Attacker's Utility

We use three parameters, l, T, and P to describe the *EDJam* attack. When the value of l closes T, our *EDJam* attack will degenerate into continuous jamming. Therefore, we assume that $l \leq \frac{1}{2}T$. An attacker's revenue (Eq. 4.3.2) is earned from the damage it inflicts on the network. We use the reduction of the network throughput ($\Delta\rho$) to denote damage. The cost (Eq. 4.3.1) refers to the energy consumed during the attack. Therefore, we obtain the formula of attacker utility as Eq. 4.3.3, which will be maximized in the following.

$$max\{\pi_a(T, t_r)\}$$

Theorem 4.3.1. *$\pi_a(T, t_r)$, with respect to T, is maximized at $T = t_r$ when $l < t_r$.*

Proof. The attacker utility π_a is

$$\pi_a = \frac{P}{T}\left(\frac{t_r}{\lceil \frac{t_r}{T} \rceil} - l\right) \qquad (4.3.6)$$

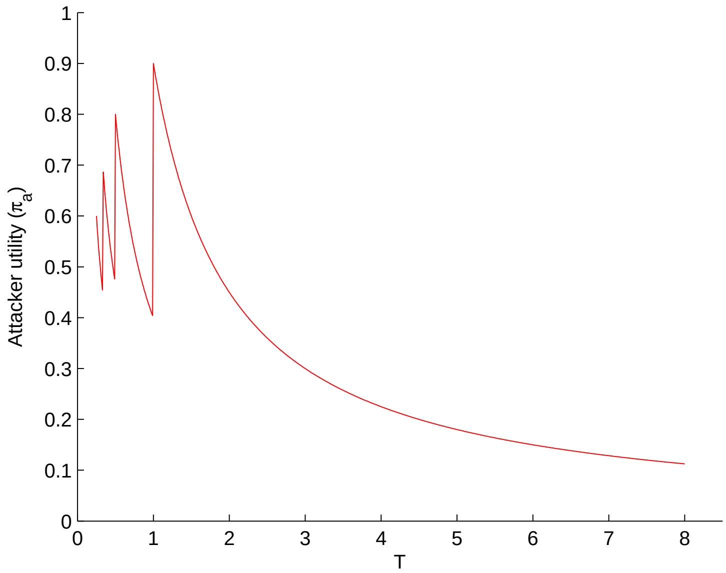


Figure 4.4: Attacker's utility function of the jamming period.

Figure 4.4 shows the curve of the attacker's utility function. When $\frac{t_r}{k} \leq T < \frac{t_r}{k-1}$ (k is an integer and $k > 0$), $\lceil \frac{t_r}{T} \rceil$ equals to k. In particular, the inequality is simplified to $T \geq t_r$ when $k = 1$. Substituting it to Eq. 4.3.6,

$$\pi_a = \frac{P}{T}\left(\frac{t_r}{k} - l\right) \qquad where \quad \frac{t_r}{k} \leq T < \frac{t_r}{k-1} \qquad (4.3.7)$$

First, we prove that $\frac{t_r}{k} > l$. When $k = 1$, it is simplified to $t_r > l$, which is the condition of the theorem. When $k \geq 2$, we got the follows,

$$l \leq \frac{1}{2}T < \frac{t_r}{2(k-1)} \leq \frac{t_r}{k}$$

.

We thus conclude that $l < \frac{t_r}{k}$.

Then, we prove that when $\frac{t_r}{k} > l$, the function of π_a, with respect to t_r, is decreasing in each slot $[\frac{t_r}{k}, \frac{t_r}{k-1})$ (k is an integer and $k > 0$). Therefore, the local maximum point in each slot is

$$(\frac{t_r}{k}, P \cdot (1 - \frac{l \cdot k}{t_r}))$$

.

Moreover, the global maximum point should be

$$(t_r, P \cdot (1 - \frac{l}{t_r}))$$

when $k = 1$.

When T equals to the global maximum point (Eq. 4.3.8), the utility function of the attacker is maximized (Eq. 4.3.9).

$$T^* = \frac{t_r}{k} = t_r \tag{4.3.8}$$

$$\begin{aligned}\pi_a^* &= \pi_a(T^*)\\ &= P \cdot (1 - \frac{l}{t_r})\end{aligned} \tag{4.3.9}$$

□

According to the analytical results (Eq. 4.3.8 and 4.3.9), we design an attack scheme to jam the network by dynamically maximizing the jammer's utility. First, a jammer senses the wireless environment periodically, to catch any changes in the network retransmission mechanism. Second, once the retransmission mechanism changes, this means that there is a new retransmission timer. To calculate the new timer, a jammer can use the method presented in Section 4.2. Finally, according to the value of the new retransmission timer and our analytical result, the jammer can choose an optimal strategy by setting the jamming period to the same value as that of the new retransmission timer. The analytical result will be verified by the simulation in Section 4.4.

4.3.3 Network Defense Strategy

The utility of network is calculated as Eq. 4.3.5. Devices can also figure out the best response (Eq. 4.3.11) and use it to defend against an attack. According to the jammer's optimal period, $T = t_r$, we can simplify the utility devices as Eq. 4.3.10.

$$\pi_d = 1 - P + \frac{1}{1 + e^{-a(t_r - t_{r_0})}} \tag{4.3.10}$$

$$max\{\pi_d(T, t_r)\} \tag{4.3.11}$$

Theorem 4.3.2. *When the following condition, $a \cdot T > 4P$, is satisfied, a device maximizes its utility function if and only if the retransmission timer t_r is set to the following optimal value,*

$$t_r = t_{r_0} - \frac{1}{a} \cdot \ln\left(\frac{a \cdot T + \sqrt{a \cdot T(a \cdot T - 4P)}}{2P} - 1\right)$$

Proof. Calculating the first order derivative of π_d with respect to t_r, we have,

$$\frac{\partial \pi_d}{\partial t_r} = -\frac{P}{T} + \frac{a \cdot x}{(1 + x)^2} \tag{4.3.12}$$

where x is given by

$$x = e^{-a(t_r - t_{r_0})}$$

Obviously, $\frac{x}{(1+x)^2} \leq \frac{1}{4}$ since $x \geq 0$. Therefore, when $a \cdot T \leq 4P$, the first order derivative of π_d with respect to t_r is always nonpositive,

$$\frac{\partial \pi_d}{\partial t_r} = -\frac{P}{T} + \frac{a \cdot x}{(1 + x)^2} \leq -\frac{P}{T} + \frac{a}{4} \leq 0$$

Therefore, the network utility is a monotonically decreasing function. Because $t_r \geq 0$, π_d achieves its maximum at $t_r = 0$ when $a \cdot T \leq 4P$.

As shown in Figure 4.5, when $a \cdot T > 4P$, with the increase of t_r, π_d decreases first, then increases, and at last decreases. There is a local minimum point t_{r_1} and a local maximum point t_{r_2} when π_d is defined in the whole positive real number field (the metric of t_r is time which will not be a negative number at all). Both t_{r_1} and t_{r_2} can be derived by assigning the first order derivative of π_d to be 0 (Eq. 4.3.13).

$$\frac{\partial \pi_d}{\partial t_r} = -\frac{P}{T} + \frac{a \cdot x}{(1+x)^2} = 0 \tag{4.3.13}$$

We have,

$$x_1 = \frac{-2P + a \cdot T - \sqrt{a \cdot T(a \cdot T - 4P)}}{2P}. \tag{4.3.14}$$

$$x_2 = \frac{-2P + a \cdot T + \sqrt{a \cdot T(a \cdot T - 4P)}}{2P}. \tag{4.3.15}$$

x is given by $x = e^{-a(t_r - t_{r_0})}$, therefore, we have,

$$t_{r_1} = t_{r_0} - \frac{1}{a} \ln \frac{-2P + a \cdot T - \sqrt{a \cdot T(a \cdot T - 4P)}}{2P}. \tag{4.3.16}$$

$$t_{r_2} = t_{r_0} - \frac{1}{a} \ln \frac{-2P + a \cdot T + \sqrt{a \cdot T(a \cdot T - 4P)}}{2P}. \tag{4.3.17}$$

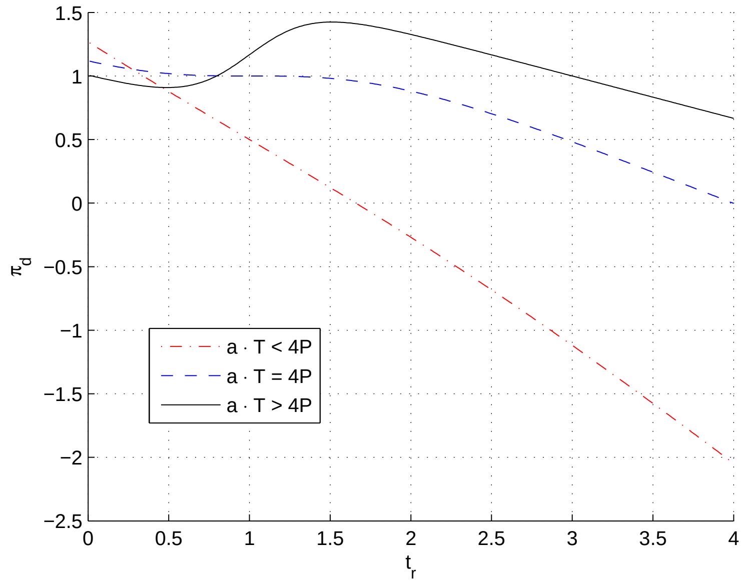


Figure 4.5: Network utility (π_d) versus retransmission timer (t_r).

When t_r is equal to the local maximum point, which is also the global maximum point (Eq. 4.3.18), the utility function of the device is maximized (Eq. 4.3.19).

$$t_r^* = t_{r_0} - \frac{1}{a} \ln \frac{-2P + a \cdot T + \sqrt{a \cdot T(a \cdot T - 4P)}}{2P} \tag{4.3.18}$$

$$\pi_d^* = \pi_d(t_r^*) \tag{4.3.19}$$

□

According to the analytical result in Eq. 4.3.18, we propose an adaptive countermeasure against *EDJam* in the IEEE 802.15.4 compliant network by dynamically selecting the retransmission timer. As illustrated in Eq. 4.3.18, a and t_{r_0} are predefined parameters for the attacker and network. The optimal retransmission timer (t_r) is a function of the jamming period (T) and packet drop rate (P), which may change with real channel conditions and is decided by the jammer. In order to study the real time packet drop rate in various environments, we have implemented the jamming signal by exploiting the *CSMA-CA* mechanism used in 802.15.4. In order to calculate the current jamming period, legitimate nodes periodically detect channel conditions to find jamming signals appearing from the jammer and record their intervals. Some legitimate nodes that detect the jamming signal will send their intervals to the coordinator. Then, the coordinator will calculate the optimal retransmission timer according to the new jamming period.

After the calculation, the coordinator will broadcast the optimal timer to the legitimate nodes in the next beacon frame. Therefore, in order to implement this dynamic retransmission, the timer value has to be added to the beacon frame for notifying all legitimate nodes to synchronize the network retransmission mechanism.

Theorem 4.3.3. *A unique Nash Equilibrium point exists if and only if* $t_{r_0} > \frac{4 \cdot P}{a}$.

Proof. According to the optimal equation (Eq. 4.3.18), we can derive the fact that t_r is a decreasing function with respect to T. As illustrated in Figure 4.6, we prove that an intersection point exists if and only if $t_{r_0} > \frac{4 \cdot P}{a}$.

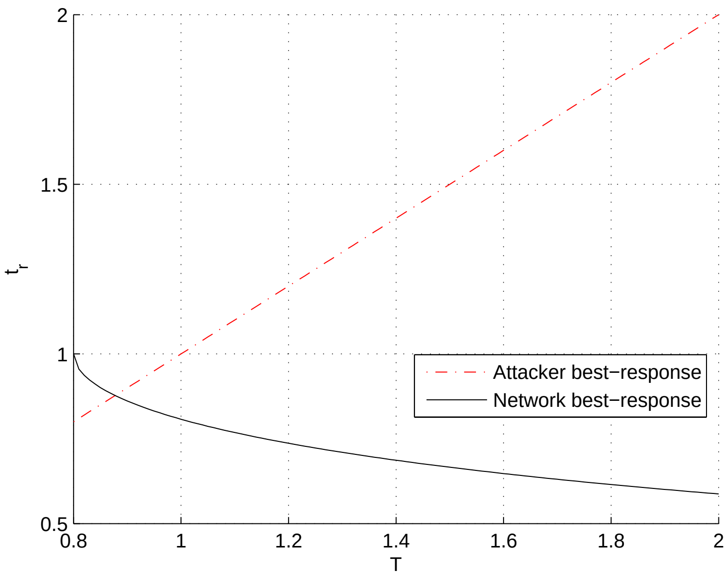


Figure 4.6: Jammers and the network's best-response curves. The intersection point is the equilibrium point.

When T is equal to its minimum value $\frac{4P}{a}$, $t_r = t_{r0}$. $t_{r_0} > \frac{4 \cdot P}{a}$, therefore, $t_r > \frac{4 \cdot P}{a}$. This means that the point $(\frac{4 \cdot P}{a}, t_r(\frac{4 \cdot P}{a}))$ is above the attacker's best-response curve. In addition, the network's best-response is a decreasing function so that we can find a point under the attacker's best-response. The network best-response is a continuous function; hence, there must be an intersection point between the attacker's and the network's best-response curves.

Similarly, if there is an intersection point between the attacker's and the network's best-response curves, the point $(\frac{4 \cdot P}{a}, t_r(\frac{4 \cdot P}{a}))$ must be over the attacker's best-response curve. If not, according to the decreasing character of the network's best-response function, there would be not any intersection point because the network's best-response curve is below the attacker's best-response curve at all times. Therefore, no intersection point would exist. □

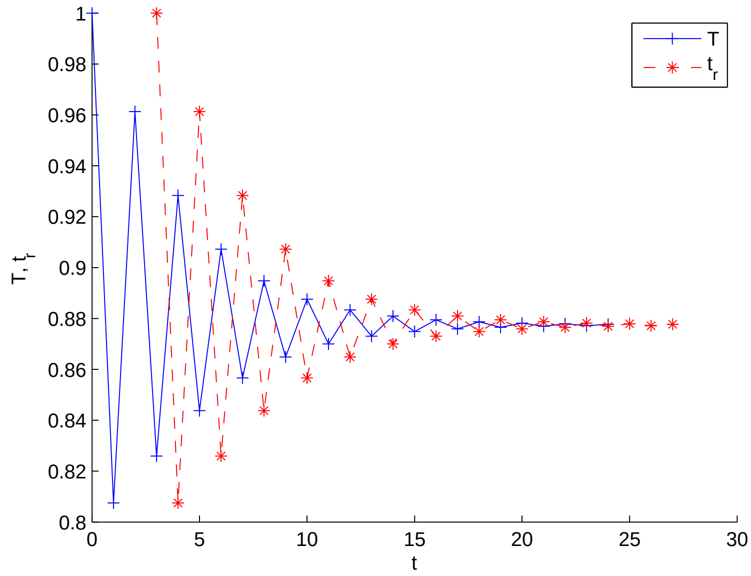
4.4 Performance and Evaluation

In this section, we show simulation results to illustrate the impact of the jamming period for attacking and retransmission mechanism for defending the jammer. Moreover, we compare our *EDJam* with continuous and random jamming, which are classic jamming attacks in wireless networks. In the simulation, we consider a simple transfer model where devices transfer data to a coordinator using beacon-enabled IEEE 802.15.4.

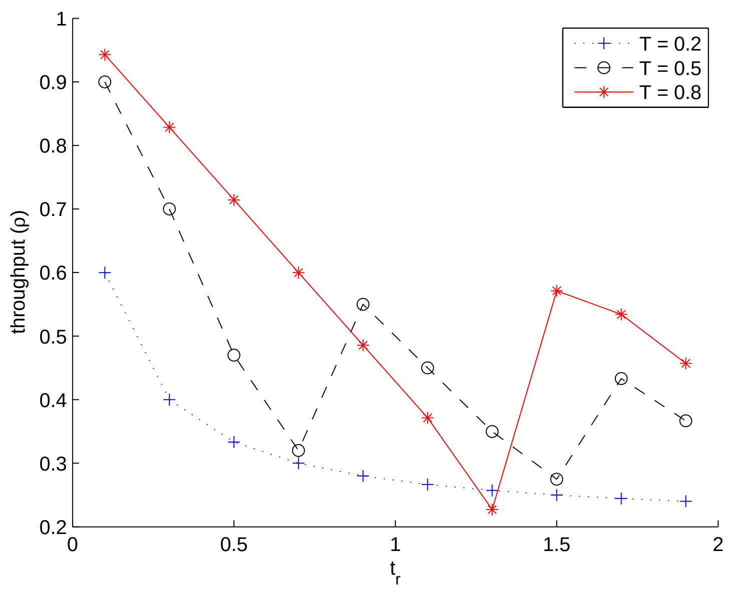
4.4.1 Impact of the Jamming Period and the Retransmission Timer

We used a homemade simulator programmed by C++ to simulate an effective dynamic jamming schedule. We have implemented the generation of jamming signals in the network composed by MicaZ motes that use 802.15.4 as their communication standard. Therefore, our simulator focused on how the jammer controls the jamming period (T) and how the network adjusts its retransmission timer (t_r) to maximize their respective utilities.

Figure 4.7 (a) shows the temporarily optimal T and t_r, versus time t. With the increase of time, T and t_r converge to an equilibrium point. Finally, the retrans-



(a) The attacker and network respectively adjust the jamming period (T) and the retransmission timer (t_r) to maximize their utilities.



(b) Network normalized throughput versus the network retransmission timer (t_r) when the attacker uses various jamming periods.

Figure 4.7: Game process and the normalized throughput.

mission timer t_r will be equal to the jamming period T because the condition of an attacker maximizing its utility is $T = t_r$. In the figure, we can see that these two parameters converge to the same point.

Figure 4.7 (b) illustrates how the network defense (t_r) impacts the normalized throughput (ρ). In this simulation, we fix the value of the jamming period T, e.g. t_{r_0}, which is the default retransmission timer in IEEE 802.15.4 and the first optimal value of T. As shown in Figure 4.7 (b), when the network uses a static strategy, a constant retransmission timer (t_r), to defend itself against the jamming, an attacker can dynamically adjust its jamming period (T) to decrease the network normalized throughput. For example, when $t_r = 0.5$, the attacker can choose $T = 0.5$ to jam the network. Therefore, in order to defend itself against this dynamic jamming schedule, the network should adjust its defense strategy correspondingly, e.g. by choosing $t_r = 1.5$. We have proposed this kind of defense in Section 4.3.

Figure 4.8 illustrates how the drop rate (P) impacts the optimal jamming period T and retransmission timer t_r. There is a point at which the two curves intersect. When P is larger than the value of the intersection point, the values of the optimal T and t_r are larger. This means that the more quickly the satisfaction of the retransmission mechanism increases, the larger the optimal values of T and t_r are. This reveals that a larger reaction sensitivity of the network leads to an increase in the equilibrium point and a decreased in the normalized throughput.

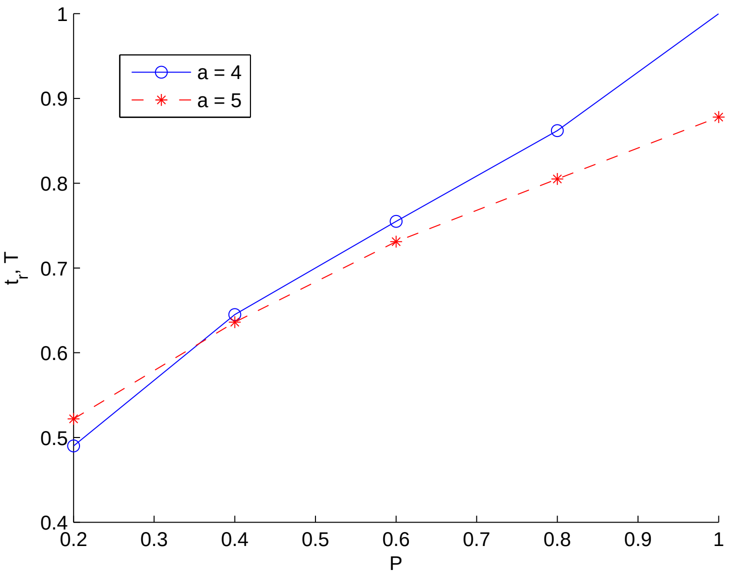
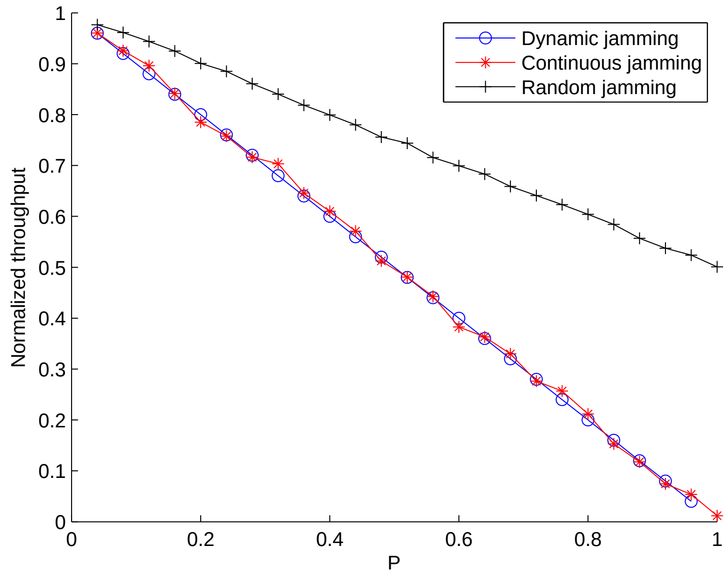


Figure 4.8: The optimal network retransmission timer (t_r) and jamming period (T) versus the drop rate during jamming P.

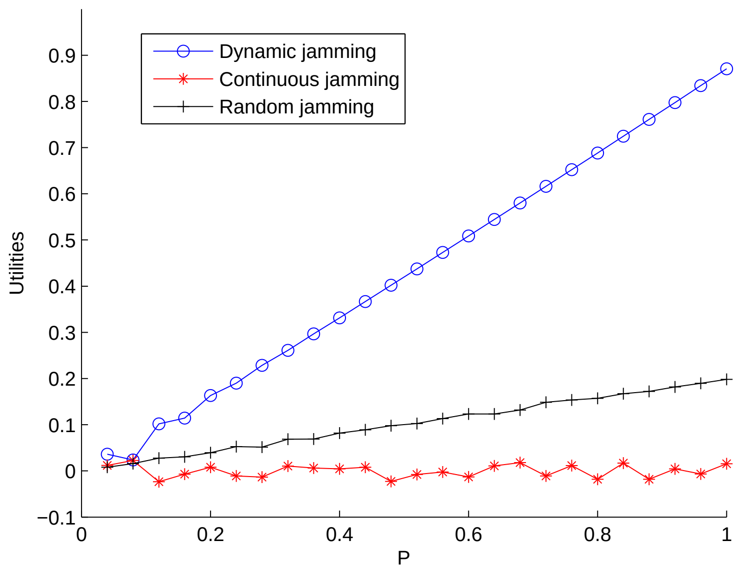
4.4.2 Comparison

Now we turn to compare the performance of our *EDJam* attack with continuous, random, and fixed-period jamming in terms of utility and normalized throughput. We will also study the difference between dynamic and static defense strategies. From the following result, we can conclude that *EDJam* obviously has a higher utility than continuous and random jamming.

Figure 4.9 (a) shows that the network normalized throughput under continuous jamming is equal to under EDJam, which is lower than under random jamming. This means that the continuous jamming can make the same damage with EDJam to the network, then EDJam, then random jamming. EDJam can force a sender



(a) Normalized throughput of EDJam, random jamming, and continuous jamming versus the jamming power (P).



(b) Utilities of EDJam, random jamming, and continuous jamming versus the jamming power (P).

Figure 4.9: Comparison among EDJam, continuous jamming and random jamming.

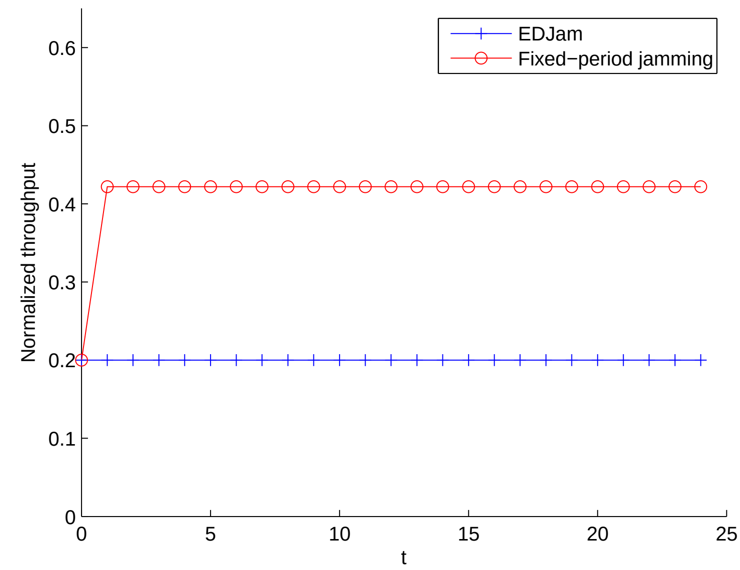
to enter the retransmission state continuously, so EDJam can cause the similar damage to a continuous jamming which emits jamming signals continuously.

Although continuous jamming can produce the same damage with EDJam, continuous jamming obviously cost more energy than EDJam. Figure 4.9 (b) shows that no matter what jamming signals strength (P) is used, the utility of an EDJam attack is higher than those of both continuous and random jamming attacks.

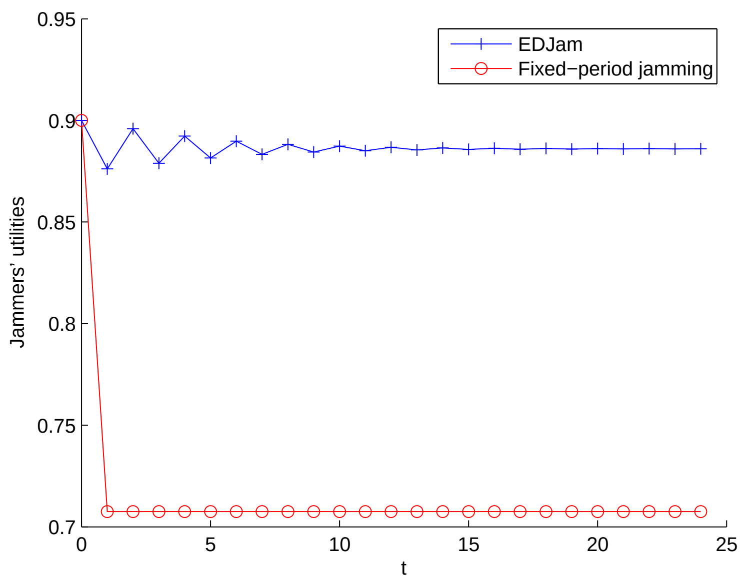
Figs. 4.10 illustrates that EDJam has higher utility and damage than fixed-period jamming. Fixed-period jamming differs from EDJam in that the jamming periods are dynamically adjusted to optimize the utility or are decided right at the beginning. Under both attacks, the network uses retransmission mechanism to defend against them. It reveals that dynamic period has higher utility than the fixed-period jamming because of the flexible strategy and optimizing process of EDJam.

4.5 Summary

In this chapter, we proposed an effective dynamic jamming strategy to attack IEEE 802.15.4-compliant wireless personal area networks. The attacker dynamically adjusts its jamming period to achieve the optimal utility, while the network will change its retransmission mechanism correspondingly to defend the jamming. A jamming strategy, where the attacker works in a jamming slot and sleeps in the



(a) Normalized throughput of EDJam and fixed-period jamming using dynamic countermeasure versus the time.



(b) Utilities of EDJam and fixed-period jamming using dynamic countermeasure versus the time.

Figure 4.10: Comparison between EDJam and fixed-period jamming.

remaining time slot, was designed in the model to attack effectively and reduce energy consumption. We formulated this model as a Stackelberg game and derived the Nash Equilibrium for the game by maximizing the utilities of network and the attacker respectively. In the simulation, the numerical results showed that the attacker and network can achieve optimal utility in the procedure of jamming and defending and will finally converge to a point of equilibrium.

Chapter 5

Energy-Efficient Uncoordinated Direct Sequence Spread Spectrum Against Jamming Attacks

In this chapter, we introduce the proposed energy-efficient uncoordinated direct sequence spread spectrum (EUDSSS) and how to defend reactive jamming attacks (e.g. the jamming scheme proposed in Chapter 3 and 4). This chapter is organized as follows. Section 5.1 briefly introduce the work. Section 5.2 presents the basis of DSSS and the jammer capability. Section 5.3 proposes a simple keyless DSSS for jamming-resistant communication. Section 5.4 proposes an efficient keyless DSSS by optimizing the tradeoff. Section 5.5 evaluates the jamming-resistant mechanism through simulations. Section 5.6 draws the conclusion.

5.1 Overview

Various jamming problems in wireless communication have been widely discussed in the previous works. The common jamming countermeasure is spread spectrum such as FHSS and DSSS. In these both technologies, jamming resistance has only possible by using the shared key to generate the sequence of frequencies or spreading codes.

However, pre-share key is not scalable to large systems (e.g. GPS and some military application) and suffers from the network dynamics problem in ad-hoc network. Meanwhile, an attacker may learn the shared key from a compromised or malicious receiver. Recent work [Strasser and Pöpper, 2008, Baird et al., 2007, Liu et al., 2010, Pöpper et al., 2009] identify the lack of keyless methods for jamming-resistant communication and propose some methods to solve this problem. An uncoordinated frequency hopping (*UFF*) [Strasser and Pöpper, 2008] is proposed to reliably establish secret key when there are jammers existing in the wireless network. Baird et al. present *BBC* algorithm [Baird et al., 2007] which uses concurrent codes in combination with UWB pulse transmissions. A major limitation of *BBC* is the number of pulses that the attacker can insert.

The Randomized Differential DSSS (*RD-DSSS*) [Liu et al., 2010] is presented to guarantee jamming-resistant wireless broadcast communication. To reduce the overhead of communication, authors use an index code to denote a sequence of spread code. However, the index code is vulnerable when the sender transmits bit '1'. Moreover, to resist reactive-jamming, authors permute all codes of the spread message and use a mapping function to reconstruct them. The computational overhead during the reconstructing procedure is large. It cannot meet the requirement of the energy-constrained network in which the overhead is an important performance metric. When design a spread spectrum technology in the energy-constrained network, the balance between the capacity of jamming-resistance and overhead introduced by the technology should be considered as one of the critical design criterion. In this chapter, we proposed an energy-efficient keyless DSSS (EUDSSS) which is suitable for energy-constrained wireless networks.

5.2 Network and Jammer Models

In this section, we present an overview of the basic DSSS technology and the jamming attacks.

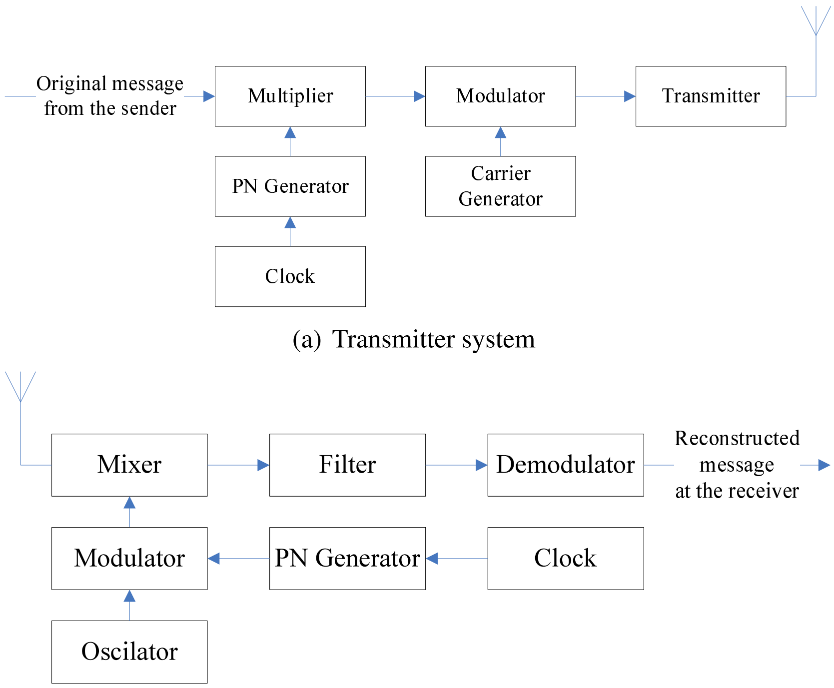
5.2.1 DSSS

In this chapter, we focus on a pair of sender and receiver that are wireless devices which can transmit and receive radio frequency signals. They reside within each other's power range. Both receiver and sender have processing and storage units, a clock, and a radio transceiving module. In the communication between the sender and the receiver, the original message is divided into several equal-length pieces to transmit and spread by DSSS technology. DSSS is a modulation technique used in digital signals transmission.

Spreading: In direct sequence spread spectrum transmissions (Figure 5.1 (a)), a sender multiplies the digital data (M) by spreading codes. The result binary bitstream is called *chips* or *spread message*. Then it uses a D/A converter to transform the chips into analog signal (baseband signal). Finally, this baseband signal is multiplied by a cosine signal to generate the radio frequency signal which will be transmitted in the wireless channel. In this chapter, we focus on the first step which is known as “spreading”. The spread code is a *pseudo-random sequence (PN)* of 1 and -1 values, at a rate much higher than that of the original signal (M), thereby spreading the energy of the original signal into a much wider band. As shown in Figure 5.2, when the sender transmits bit ‘1’, it multiplies ‘1’ by spreading code and generates the spread signal (the same with spread code when original bit is ‘1’). When the sender transmits bit ‘0’, it multiplies ’-1’ by spreading code and generate the spread signal (reversion of the spreading code).

De-spreading: The received signal resembles white noise. However, this noise-like signal can be used to exactly reconstruct the original data at the receiving end, by multiplying it by *the same pseudo-random sequence*. This process, known as “*de-spreading*”, mathematically constitutes a correlation of the transmitted spread code with the spread code that the receiver believes the transmitter is using. As shown in Figure 5.2, when the destination receives a spread signal, it multiplies the bitstream of the spread signal by spreading code and generate the original bit. It is ‘1’ when the spread signal is the same with spreading code. Otherwise, it is ‘0’.

For de-spreading to work correctly, the transmitted and received sequences



(a) Transmitter system

(b) Receiver system

Figure 5.1: A simple communication model of DSSS.

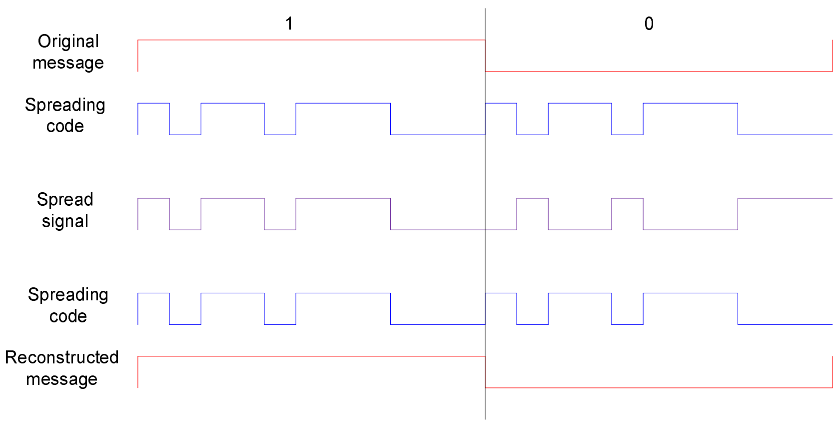


Figure 5.2: A spreading and de-spreading example in DSSS.

must be synchronized. This requires the receiver to synchronize its sequence with the transmitter's sequence via some sort of timing search process. Meanwhile, the receiver must synchronize the pseudo-random sequence (generate a same pseudo-random sequence with the one used by sender) to reconstruct the message.

5.2.2 Jamming Attacks

In wireless networks, communications among devices will be interfered when noises arise because of the shared medium, air. By exploring this vulnerability, attackers can destroy the communication. One of the schemes is jamming attacks which intend to corrupt the normal transmitted packets by emitting malicious noises (interference signals). In this chapter, we assume the capability of jammer as followings:

1. A jammer can jam messages by transmitting high-power interference signals that corrupt the original signal. We use *interference probability* to denote this capability in our model. When a jammer emits malicious signals to the air, some transmitted bits may be corrupted and the received bits are incorrect in the destination. Therefore, we define the bit error ratio at destination to be the interference probability $P_i = \frac{\text{the number of error bits}}{\text{the number of total transmitted bits}}$.

2. A jammer can insert message.

3. A jammer can eavesdrop on the wireless channel used by the sender and receiver.

4. A jammer can identify the spreading code used in traditional DSSS.

5. A jammer cannot successfully jam the transmission when he does not know the spreading code. He just uses the fixed power to jam the channel by randomly choosing a spread code.

5.2.3 Jamming Resistance

In wireless communication, jammers use narrowband interference signals to corrupt the transferring packet and increase the value of *BER*. Jamming resistance is one of the performance metrics of network capability to defend interference signals. We use *Bit Error Rate (BER)* to denote this metric. Lower *BER* means high jamming-resistance of the spread spectrum technology. In digital communication, the number of bit errors is the number of received bits of a data stream over a communication channel that has been changed due to noise, interference, distortion or bit synchronization errors. The *BER* is the ratio of the number of bit errors to the total number of transmitted bits during an observation time slot.

It is a dimensionless and normalized performance metric, often expressed as a percentage number. As an example, assume the transmitted bit sequence: 1001110100, and the following received bit sequence: $1\underline{1}01\underline{0}101\underline{11}$. The number of bit errors (the underlined bits) is in this case 4. The *BER* is 4 incorrect bits divided by 10 transferred bits, resulting in a *BER* of 0.4 or 40%.

If the wireless network has used the spread spectrum technology, although jammers transmit on the same channel but with a different spread code (or no spread code at all) from the sender, the receiver still correctly de-spread the message. Meanwhile, the value of *BER* would not increase when a jammer present. This property make the spread message can resist jamming attacks. In this chapter, by using a spread spectrum technology, we assume that the network require that the value of *BER* is smaller than a threshold α. The length of spread code could affect the value of *BER*. Therefore, a proper design of the length can meet the requirement of *BER* value.

Jammers use narrowband interference signals to corrupt the transferring packet. The resulting effect of enhancing signal to noise (e.g. interference signals) ratio on the channel is called process gain. This effect can be made larger by employing a longer PN sequence and more chips per bit, but physical devices used to generate the PN sequence impose practical limits on attainable processing gain. In [], the capability (processing gain) of narrowband interference suppression is calculated as $G_p = \frac{R_c}{R_d}$, where R_c denotes the pseudo code rate and R_d denotes the baseband chip rate. For DSSS, assuming that one baseband chip is spread by one cycle pseudo codes and the cycle of pseudo code is T_s, therefore the processing gain is $G_p = T_s$.

If an undesired transmitter (e.g. jammer) transmits on the same channel but with a different PN sequence (or no sequence at all), the de-spreading process results in no processing gain for that signal. This effect is the basis for the code division multiple access (CDMA) property of DSSS, which allows multiple transmitters to share the same channel within the limits of the cross-correlation properties of their PN sequences. Likewise, this property make the spreading message can resist jamming attacks.

5.2.4 Overhead

As using DSSS, the cost of device is affected by three factors. The first factor is computational overhead which includes the energy and time used for spreading, de-spreading and modulating the message. The second factor is storage overhead which includes the space used to store the spreading code set. The third factor is communication overhead which includes the additional bits by using the spread spectrum technology. The main factor impacting the cost of DSSS is the communication overhead. In this chapter, we show that a better keyless DSSS design can reduce the cost considerably without any significant change in the traditional DSSS mechanism.

5.3 A Simple Uncoordinate DSSS

In this section, we introduce a simple uncoordinate DSSS. It does not require a pre-share secret key. However, its overhead is more than twice as traditional DSSS if they use the same length of spread code. An improved uncoordinate DSSS would be proposed in the next section to increase its efficiency with the requirement of jamming-resistance.

5.3.1 A Simple Uncoordinate DSSS

In the simple uncoordinate DSSS, a spreading code set is used to store the spreading codes. The code sequence set is denoted as $S = s_1, s_2, ..., s_m$ where s_i ($i \in 1...m$) is a spreading code sequence with fixed length l_s ($s_i = c_{i1}, c_{i2}, ..., c_{il_s}$). We define a correlation function on this set. The correlation of two spreading code sequences (s_i and s_j) is defined to be $s_i \cdot s_j = (c_{i1} * c_{j1} + c_{i2} * c_{j2} + ... + c_{il_s * c_{jl_s}})/n$. To calculate the correlation, we replace "0" by "-1" in the formula. We list the requirements of spreading code set as followings.

• Two different spreading codes in the set should have low correlation. It means that if s_i and sj belong to S and $s_i \neq s_j$, $s_i \cdot s_j$ will be lower than a pre-defined threshold β. If the sender is intent to send a bit "1", he will use *two different spreading codes* to encode it.

• Two identical spreading codes in the set should have high correlation. It means that if s_i and sj belong to S and $s_i = s_j$, $s_i \cdot s_j$ will be higher than a pre-defined threshold β. As in traditional DSSS, if the sender is intent to send a bit "0", he will use *two identical spreading codes* to encode it.

• The spreading code set is known to the public. In the proposed SUDSSS, there is not pre-share key to synchronize the spreading code. In SUDSSS, we use the "correlation" to encode the original bit stream. A sender who intends to transmit a message will select a pair of codes from the spreading code set to encode one bit of the message.

We use an example to illustrate our SUDSSS. In the example, a sender sends a 5-bit message "10110" to a receiver. The sender randomly selects codes s_1, s_2, s_6, s_8 and s_5 from the spreading code set. The sender uses two identical spread codes s_1 to encode the first bit "1". The second bit "0" will be encoded by s_2 and another different spreading code (e.g. s_3) from the set. The third, fourth and fifth bits are encoded similarly. Finally, the original bit stream "10110" is encoded as $s_1 \parallel s_3 \parallel s_6 \parallel s_8 \parallel s_9 \parallel s_1 \parallel s_2 \parallel s_6 \parallel s_8 \parallel s_5 \parallel$. The first half of this encoded message are the codes that randomly choose from the spreading set. The other half are the codes that carefully selected by the sender according to the encoding method described in the previous paragraph.

When a receiver receives the encoded message, it first computes the correlation between the corresponding codes in the first half and the latter half of the encoded message. Then the receiver compares the correlation with the threshold. If the correlation is lower than α, the receiver will decode it as 1. Conversely, if the correlation is higher than β, the receiver will decode it as 0. In the following, we present more details of the SUDSSS.

Spreading: Let $m = m_1 \parallel m_2 \parallel ... \parallel m_l$ denote the message transmitted by the sender. First, the sender randomly selects a code sequence from S. Let $L = s_1 \parallel s_2 \parallel ... \parallel s_l$ denotes the selected code sequence. Then the sender use this selected code sequence to generate the spread message. We use $MS = ms_1 \parallel ... \parallel ms_l \parallel ms_{l+1} \parallel ... \parallel ms_{2l}$ to denote the spread message where $ms_1 \parallel ... \parallel ms_l = s_1 \parallel ... \parallel s_l$ and use the following formula to generate $ms_{l+i}(i = 1..l)$:

$$\begin{cases} ms_{l+i} = s_i & when \quad m_i = 1 \\ ms_{l+i} = s_j(j \neq i) & when \quad m_i = 0 \end{cases}$$

De-spreading:) Assume the received message is $MSR = msr_1 \parallel ... \parallel msr_l \parallel msr_{l+1} \parallel ... \parallel msr_{2l}$. The receiver needs to calculate the following correlations to de-spread the received message: $(msr_1 \cdot msr_{i+1}), (msr_2 \cdot msr_{i+2}), ..., (msr_l \cdot msr_{2l})(i = 1...l)$. Let $MD = md_1 \parallel md_2 \parallel ... \parallel md_l$ denote the de-spreading output (reconstructed message). According to the following formula we can reconstruct the message:

$$\begin{cases} md_i = 1 & when \quad (mdr_i, mdr_{l+i}) \geq \beta \\ md_i = 0 & when \quad (mdr_i, mdr_{l+i}) < \beta \end{cases}$$

where β is a threshold to decide the correlation between two codes. It can impacts the BER at receiver of SUDSSS. Another main factor, length of spread code, which can impact the BER has been discussed in Section 5.2.

There are four cases of SUDSSS.

- ***Case 1*** $m_i = 0, md_i = 0 \Leftrightarrow (ms_i \cdot ms_{i+l}) = 0$ and $(msr_i \cdot msr_{i+l}) < \beta$

- ***Case 2*** $m_i = 0, md_i = 1 \Leftrightarrow (ms_i \cdot ms_{i+l}) = 0$ and $(msr_i \cdot msr_{i+l}) \geqslant \beta$

- ***Case 3*** $m_i = 1, md_i = 0 \Leftrightarrow (ms_i \cdot ms_{i+l}) = 1$ and $(msr_i \cdot msr_{i+l}) < \beta$

- ***Case 4*** $m_i = 1, md_i = 1 \Leftrightarrow (ms_i \cdot ms_{i+l}) = 1$ and $(msr_i \cdot msr_{i+l}) \geqslant \beta$

In Case 1 and 4, the receiver reconstructs the bit correctly ($m_i = md_i$). However, in Case 2 and 3, the receiver reconstructs an error bit ($m_i \neq md_i$). In the following, we will use the provided system parameters to calculate the *bit error rate* and derive the optimal length of spread code.

Given additive Gaussian noise, a proper value of the threshold β has been provided in [Liu et al., 2010]. Figure 5.3 illustrates the communication framework of SUDSSS.

5.3.2 Optimal Value of l_s

The total error bits should include Case 2 and 3 under jamming attack. The capacities of jammer have been given in Section 5.2. A jammer can successfully corrupt a bit only if he uses the same spread code with the sender. The sender randomly chooses his spread code. Obviously, guessing the spread code is beyond the probability of the jammer described in Section 5.2. The jammer just can jam the channel by randomly guess the spread code. The length of the spread code is l_s. For each bit of spread code, the jammer can correctly guess the bit by probability of 0.5. Therefore the theoretical value of *BER* is calculated as following:

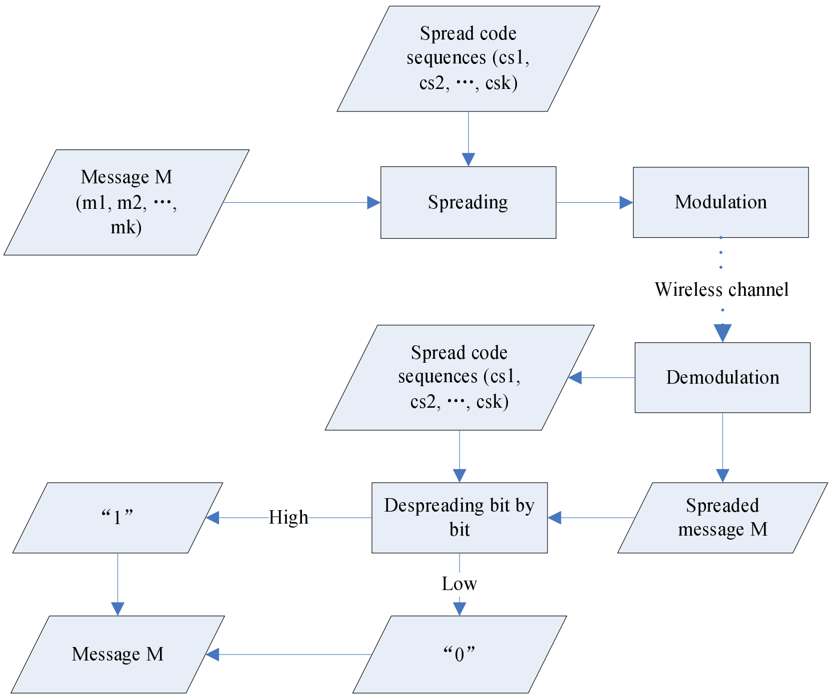


Figure 5.3: Spreading and de-spreading of *SUDSSS*.

$$BER_T = (\frac{1}{2})^{l_s} \tag{5.3.1}$$

Let the BER be smaller and equal to the threshold α. We can derive that the optimal value of l_s is $\log_2 \frac{1}{\alpha}$. From Figure 5.4, we can learn that more correct received bits require longer spread code. Meanwhile, it means more overhead of communication to transmit the spread message. Thus, for an efficient system, this optimal value is not the goal. We should consider the overhead when the SUDSSS is applied. Intuitively, a longer spreading code can provide more jamming resistance. However, when a longer spreading code is used to spread the message, a device spends more power on de-spreading and storing the code. In the following section we design an efficient keyless DSSS by optimize the tradeoff between jamming-resistance and overhead.

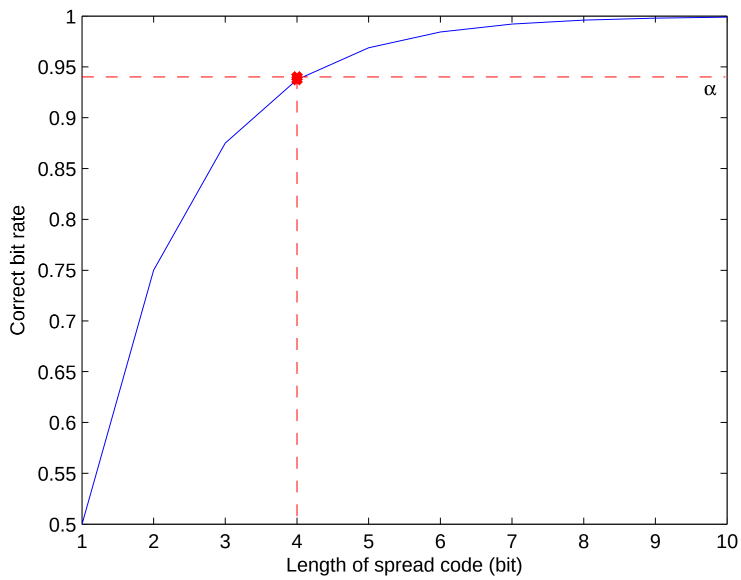


Figure 5.4: Bit error rate versus spread code length.

5.4 An Efficient Uncoordinate DSSS

In this section, we improve the simple keyless DSSS by optimizing the tradeoff between resistance and overhead. The improved keyless DSSS can balance the capacity and cost. We call it efficient uncoordinate DSSS (*EUDSSS*).

5.4.1 Tradeoff Between Resistance and Overhead

In this chapter, tradeoff is defined to be the ratio of jamming-resistance to overhead. Jamming-resistance is the capability that the spread spectrum technique resists to interference (jamming) signals. It is represented by the *correct bit rate*.

Overhead is given to one major aspect, communication.

Resistance: First, we calculate the *BER* while using SUDSSS. As described in Section 5.2, a jammer can successfully corrupt a bit only if he uses the same spread code. The length of spread code is l_s. The jammer randomly generates the spread code. Thus, the probability that the spread code generated by the jammer is the same with the spread code used by the sender is $\frac{1}{2}^{l_s}$ which is the value of *BER*. Then, therefore, the correct bit rate (jamming-resistance) is calculated as Equation 5.4.1.

$$R = 1 - (\frac{1}{2})^{l_s} \tag{5.4.1}$$

Overhead: Overhead is defined to be the increased number of packet bits due to SUDSSS. In this chapter, to normalized this metric and make it dimensionless as well as jamming-resistance, the overhead is calculated as the number of increased packet bits divided by the total number of spread bits in Equation 5.4.2.

$$V = (2l_s - 1)/2l_s \tag{5.4.2}$$

Tradeoff: Tradeoff is defined to be the ratio of jamming-resistance to the overhead as following.

$$T = \frac{R}{V} = \frac{2l_s - 1}{2l_s(1 - (\frac{1}{2})^{l_s})} \tag{5.4.3}$$

5.4.2 Optimizing the Tradeoff

From the analysis, we can learn that shorter spread code will lead to larger tradeoff. However shorter spread code will reduce the capacity of EUDSSS to resist interference signals. The analysis results show that there is a proper length of spread code to balance the capacity and cost. According to this analysis result, we design an efficient uncoordinate DSSS (EUDSSS) by optimizing the tradeoff.

A smaller value of l_s means a lower communication overhead. Meanwhile, it means a weaker resistance. We cannot choose the value of l_s arbitrarily. It must satisfy the requirement $R \geqslant (1 - \alpha)$. Subject to this constraint, we show that the values of l_s and T cannot be minimized and maximized simultaneously. The choice of l_s represents a resistant-overhead tradeoff. In this section, we maximize the tradeoff T under the constraint like following:

$$max\{T\} \qquad\qquad s.t. \qquad R \geqslant (1 - \alpha)$$

The curve of T has been shown in Figure 5.5. When $\alpha < \frac{1}{32}$ is satisfied, the constraint of resistance can be simplified as $l_s \geqslant \log_2 \frac{1}{\alpha} > 5$. Hence, the optimal value of l_s and the corresponding function value of T are given as followings.

$$\begin{cases} l_s^* = \log_2 \frac{1}{\alpha} \\ \\ T^* = \frac{1-\frac{1}{2}\log_\alpha 2}{1-\alpha} \end{cases}$$

When $\alpha \geqslant \frac{1}{32}$ is satisfied, we can prove that when $l_s < 5$, T is a increasing function and when $l_s > 5$, T is a decreasing function. Therefore, the optimal value of l_s and the corresponding function value of T is given as followings.

$$\begin{cases} l_s^* = 5 \\ T^* = 1.067 \end{cases}$$

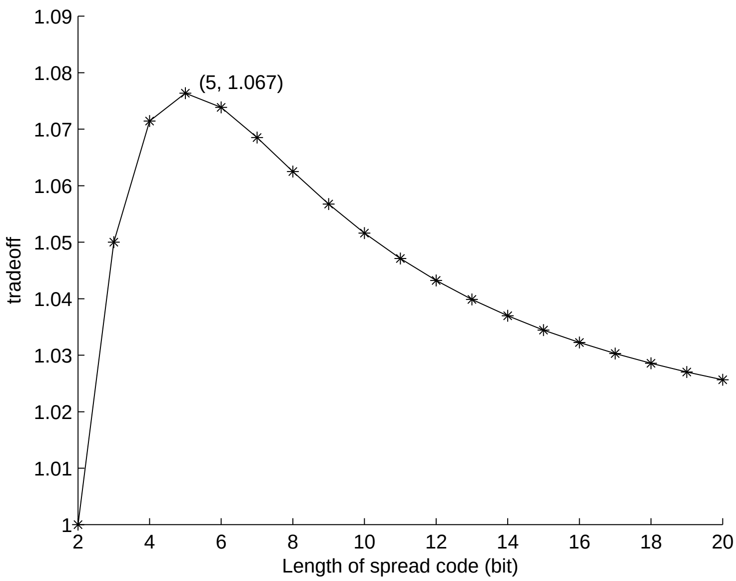


Figure 5.5: The theoretical tradeoff versus the length of spread code.

According to the requirement of resistance in the system, we can design an efficient uncoordinate DSSS by setting the length of spread code as the optimal value.

Similar to [Liu et al., 2010], EUDSSS is vulnerable to reactive jamming attackers. A reactive jammer can first observe the first f codes of transmitted message and computes the correlations between this f codes and the first f codes of all code sequence. The code sequence which can produce the highest correlation is probably the one selected by the sender. Then the jammer will use the derived code sequence to spread a fake message for jamming the remaining message.

5.4.3 Defending Reactive Jamming

We propose our final solution that addresses the limitations of the EUDSSS by inserting the selected code right after the corresponding spread code. It is very simple but effective and easy to launch. In [Liu et al., 2010], the sender permutes all codes of the spread message. Since the sender uses an index code instead of a sequence of selected codes, the sender cannot readjust selected codes. However, in our proposed scheme, we transmit the sequence of selected codes with the optimal length by involving it in the transmitted message, so that the sender can insert the spread code of the sequence right after each corresponding spread code. In this way, although jammers may derive the code, the selected code has been transferred already when jammers spread a fake message using the derived code to jam the transmission of the following selected code. Thus, the jamming attack fails to corrupt the transmitted message. Figure. 5.6 show an example of our simple solution to address the limitation against the reactive jamming. In the example, a sender transmits a 5-bit message "10110" to a receiver. Similar to the example in Section 5.3, bit "1" is encoded by two identical spread codes and bit "0" is encoded by two different spread codes. The sender randomly chooses codes s_1, s_3, s_6, s_8 and s_9 from the spreading code set. The sender uses s_1 twice to encode the first bit "1". The second bit "0" will be encoded by s_3 and another different spreading code from the set. The third, fourth and fifth bits are encoded similarly. Finally, the original bit stream "10110" is encoded as $s_1 \parallel s_3 \parallel s_6 \parallel s_8 \parallel s_9 \parallel s_1 \parallel s_2 \parallel s_6 \parallel s_8 \parallel s_5 \parallel$. The first half of this encoded message are the codes that randomly choose from the spreading set. The other half are the codes that carefully selected by the sender according to the encoding method described in the previous paragraph.

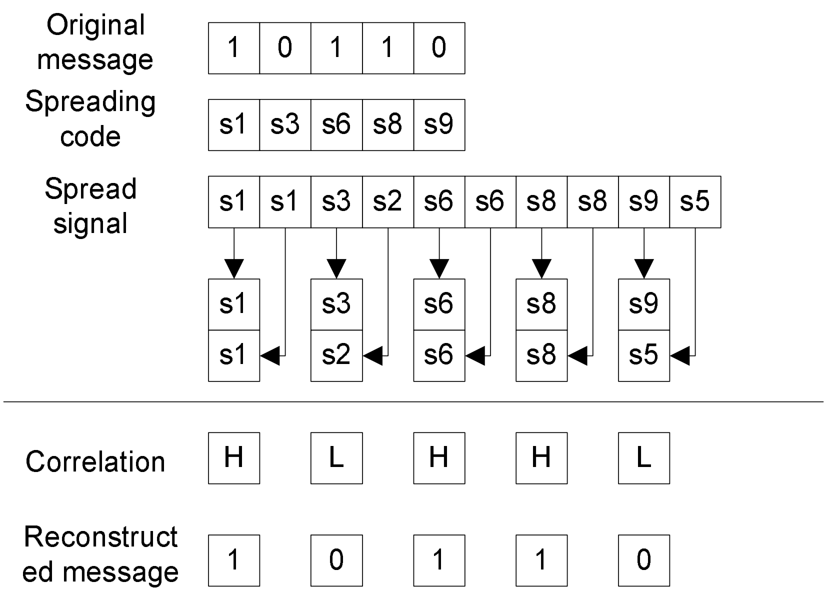


Figure 5.6: An example to resist the reactive jamming attack.

In the solution [Liu et al., 2010] proposed by Liu et. al, the sender permutes all codes of the spread message to reduce the correlation and select the spread codes from various subset of code sequence set with fixed size. The receiver needs to reconstruct the permuted codes and identify the spread code used by the sender. As shown in [Liu et al., 2010], these two procedures would cost the main overhead of computation, which cannot afford to cost in energy-constrained networks. In our solution, we introduce the overhead of communication which is acceptable since low-cost transmission is usually used in energy-constrained networks. The capability of computation is not the main function of the battery-operated device.

5.5 Simulations

In this section, we show simulation results to illustrate the impact of spreading code length for overhead and *Bit Error Rate (BER)*. In the future work, we will compare the performance of the proposed *EUDSSS* with the most related work. These simulations are carried out under various different sets of system parameter values.

5.5.1 Simulations Setting

We use matlab to simulate the SUDSSS and EUDSSS. In the simulations, the jammer can corrupt the bit only if he uses the same spread code with the sender. The sender transmit 100 messages to a receiver and a jammer appears between them to emit random interference signals to corrupt the transmitted bits. A message is composed of 2^{17} bits and is divided into 2^{13} sub-messages and each of them contains 16 bits.

5.5.2 Bit Error Rate

In the theoretical analysis, we assume that the jammer has enough power to launch the jamming attack when the jammer knows the spread code used by the sender. Therefore, the interference probability (J) is 100% in this case. Interference probability is defined to be the ratio of the number of corrupted bits to the number of transmitted bits when the jammer presents in the network without any countermeasure. However, the jammer may not corrupt the whole message due to its constrained capability. We use various interference to study the relation between length of spread code and bit error rate. Figure 5.7 shows that when the length of spreading code increase, the bit error rate will decrease with various interference probabilities. And when the length of spreading code increase up to a threshold, 15 bits, the increasing length lead to a little bit decreasing of BER. Simultaneously, when the length exceeds this threshold, the impact from the capability of jammer is smaller and smaller.

Figure 5.8 (a) shows the theoretical and simulation results of *BER* versus spread code length. *BER* increases when the length of spread code becomes longer. Meanwhile, transmitted bits will increase because the length of spreading message also increases. Thus the overhead of communication will also increas-

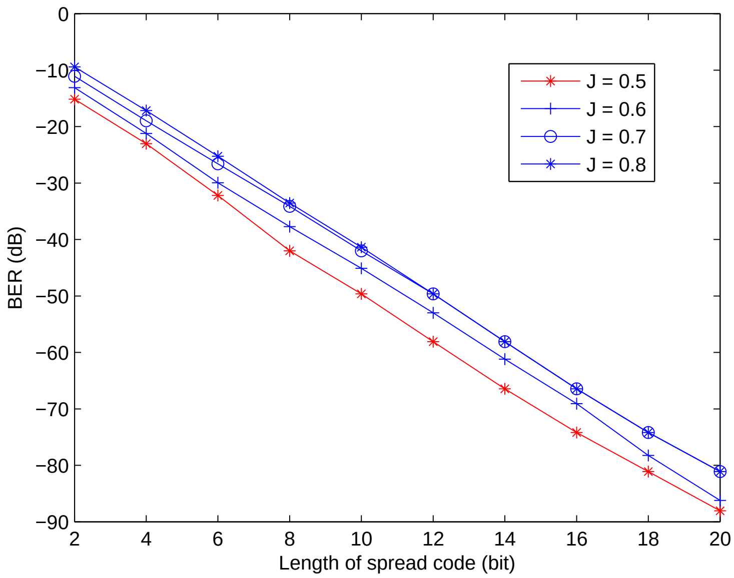
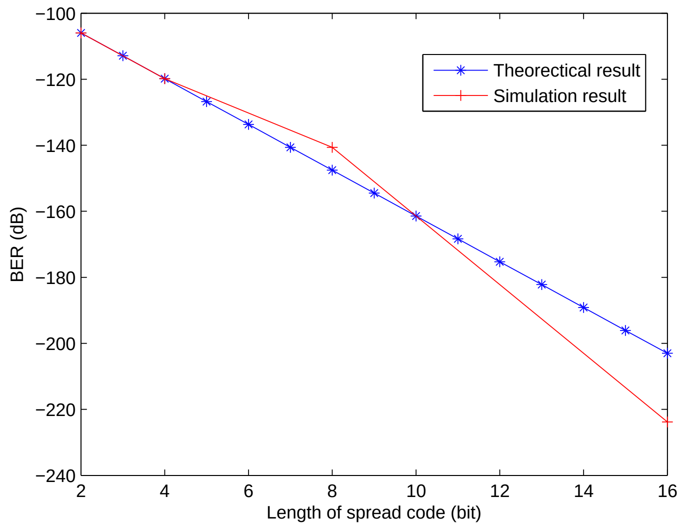


Figure 5.7: Length of spreading code versus bit error rate under various jammer capabilities.

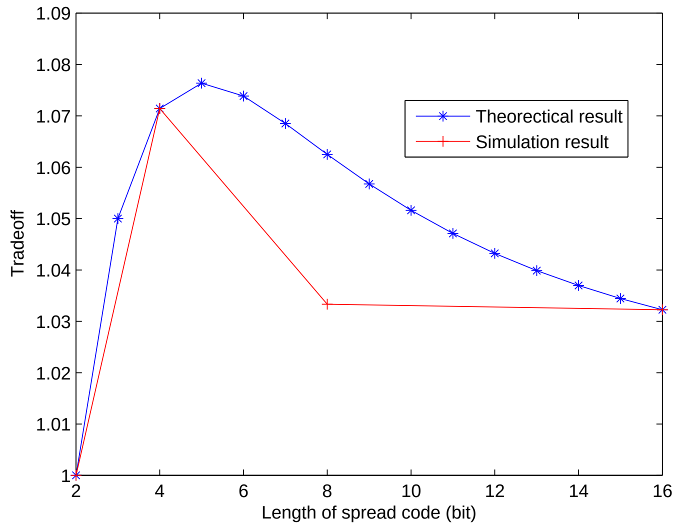
ing. It means that a longer spread code is not better for the energy-constrained network. In the simulations, we also observe the communication overhead by counting the additional bits.

Figure 5.9 shows the normalized overhead versus the length of spread code. When the length of spread code increases, the overhead will increase.

Therefore, we have the theoretical and simulation results of tradeoff in Figure 5.8 (b). From the trend of the curves, we can learn that the optimal value of l_s is close to 5. In our simulations, we just have four values of l_s (2, 4, 8, 16) which are not sufficient to illustrate the trend of tradeoff. In the future work, we will create more spread code set with various values of l_s.



(a) Theoretical and simulation results of BER versus spread code length.



(b) Theoretical and simulation results of tradeoff versus spread code length.

Figure 5.8: Comparison between theoretical and simulation results.

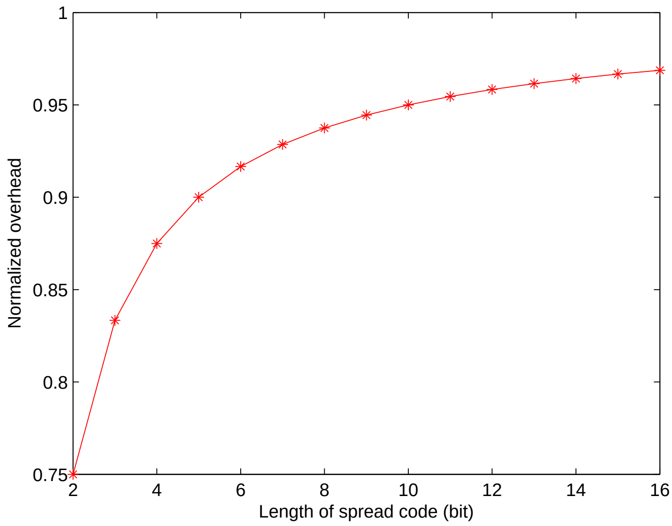


Figure 5.9: Normalized overhead versus the length of spread code.

At last, we compare our solution with the RD-DSSS proposed in [Liu et al., 2010]. As shown in Figure. 5.10, the communication overhead of RD-DSSS is always smaller than EUDSSS since RD-DSSS uses the index code to replace the code sequence. However, the communication overhead can be tolerate in the network composed of battery-operated devices since the communication volumes are small in this network.

As shown in Figure. 5.11, the computational overhead of RD-DSSS is more than ten times of EUDSSS when the length of spread code closes to the optimal value. We know that the computational capability of a battery-operated device (e.g. MicaZ) is not power enough to handle the complicated calculation and it need much time and energy to finish the computational task.

Moreover, as shown in Figure. 5.12, the storage overhead of our solution is

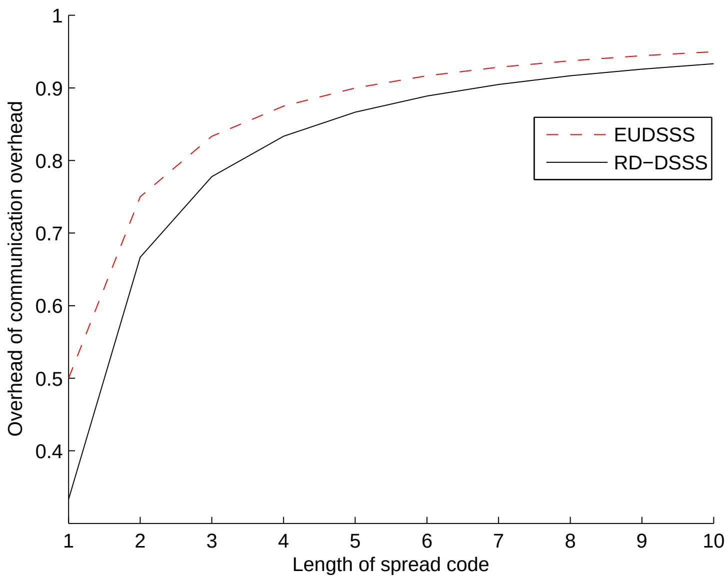


Figure 5.10: The communication overhead with respect to the length of spread code.

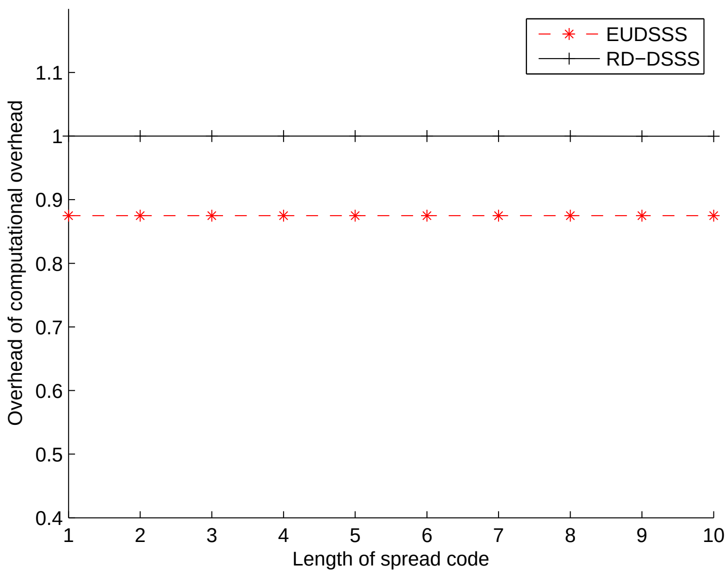


Figure 5.11: The computation overhead with respect to the length of spread code.

much smaller than the RD-DSSS. It can reduce the cost of devices manufacture and make the device more suitable for widespread deployment in some application.

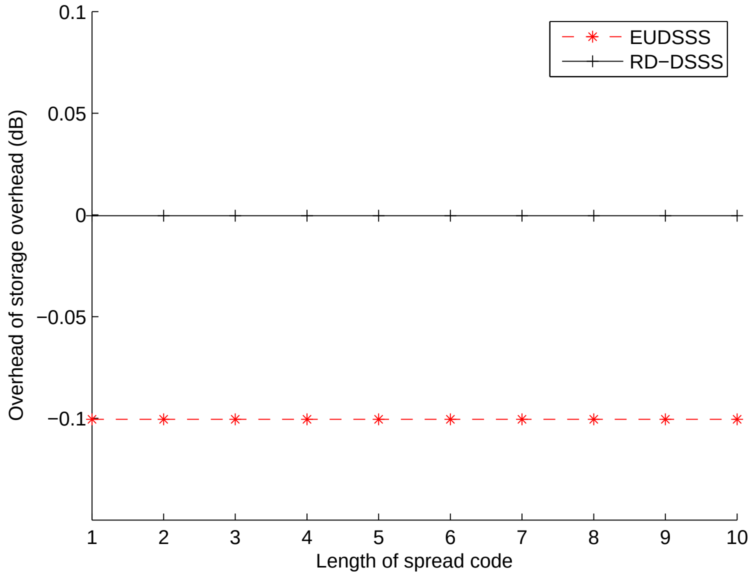


Figure 5.12: The storage overhead with respect to the length of spread code.

5.6 Summary

In this chapter, we first propose a simple uncoordinate DSSS (*SUDSSS*) scheme to enable reliable communication in the presence of jammers without a pre-share secret key. *SUDSSS* encodes each bit of message by exploiting the correlation of spread codes which are randomly selected from a public spread code set. Then, by optimizing the tradeoff between correct bit rate and overhead, we propose an efficient uncoordinate DSSS (*SUDSSS*) to balance the robustness and cost which is an important performance metric for energy-constrained devices in the network.

Chapter 6

Conclusions and Future Works

In this chapter, we first summarize the works included in this thesis. Then, we discuss the future directions of our current work.

6.1 Conclusions

Jamming attack is one of the powerful security threats in wireless networks. In this thesis, we first propose an energy-efficient reduction of quality attack against coordinators to prevent them from receiving normal packets from neighbors. In this attack, to launch an energy-efficient attack, attackers can adjust the parameter (e.g. period of generating burst traffic) to minimize the attack potency. By this means, attacker can make the maximum damage per unit cost. Moreover, we design a feasible strategy to implement burst traffic by exploiting the CSMA-CA mechanism in an IEEE 802.15.4 compatible sensor network.

Then we present an effective dynamic jamming attack *(EDJam)* in an 802.15.4-

compliant *WPAN*. Stackelber game is introduced to formulate the dynamic procedure of competition between the *EDJam* attacker and defending networks. In this game, The attacker dynamically adjusts its jamming period to achieve the optimal utility, while the network will change its retransmission mechanism correspondingly to defend the jamming. A jamming strategy, where the attacker works in a jamming slot and sleeps in the remaining time slot, was designed in the model to attack effectively and reduce energy consumption. We formulated this model as a Stackelberg game and derived the Nash Equilibrium for the game by maximizing the utilities of network and the attacker respectively. In the simulations, the numerical results showed that the attacker and network can achieve optimal utility in the procedure of jamming and defending and will finally converge to a point of equilibrium.

Finally, we propose an energy-efficient uncoordinate DSSS (*EUDSSS*) to enable reliable communication in the presence of jammers in the energy-constrained network. EUDSSS encodes each bit of message by exploiting the correlation of spread codes which are randomly selected from a public spread code set. Then, by optimizing the tradeoff between correct bit rate and overhead, we propose an efficient uncoordinate DSSS to balance the robustness and cost which is an important performance metric for energy-constrained devices in the network. The simulation results show that EUDSSS can achieve adequate reliability with small overhead.

6.2 Future Works

Our research in this thesis mainly focused on jamming attacks in energy-constrained networks. In the future work, we hope to improve the proposed approaches and to investigate related research directions.

One issue that deserves further study is the need to change the assumption of coordinator position. In our current models, the attacker knows the position of coordinator and launches our energy-efficient RoQ attack against the coordinator to degrade the network throughput. If the attacker doesn't know the position of coordinator, how can it finds out the coordinator or changes the strategies to launch an energy-efficient jamming attack?

An effective countermeasure against *EERoQ* would be implemented in the future work by adjusting the retransmission mechanism. A new retransmission mechanism will be established in the network through the beacon (the first slot of superframe). More information (e.g. timer) will be added into the beacon to notify all the nodes in the network.

In the future work, we will compare the performance of our attack schemes with more previous jamming attacks.

Bibliography

[Adamy and Adamy, 2004] Adamy, D. L. and Adamy, D. (2004). Ew 102: A second course in electronic warfare. *Artech House, Inc.*

[Baird et al., 2007] Baird, L. C., Bahn, W. L., Collins, M. D., Carlisle, M. C., and Butler, S. C. (June 2007). Keyless jam resistance. *Proceedings of the IEEE information Assurance and Security Workshop*, pages 143–150.

[Cagalj et al., 2005] Cagalj, M., Ganeriwal, S., Aad, I., and Hubaux, J.-P. (March 13-17 2005). On cheating in csma/ca ad hoc networks. *In Proceedings of the IEEE INFOCOM 2005 Conference*.

[Deng et al., 1999] Deng, J., Han, R., and Mishra, S. (1999). Defending against path-based DoS attacks in wireless sensor networks. *Proc. International Symposium on Spatial Databases (SSD)*, pages 147–164.

[Fudenberg and Tirole, 1993] Fudenberg, D. and Tirole, J. (1993). Game theory. *MIT Press*.

[Guirguis et al., 2004a] Guirguis, M., Bestavros, A., and Matta, I. (2004a). Adaptation=vulnerability: Under RoQ attacks. *Poster in ACM SIGCOMM 2004*.

[Guirguis et al., 2004b] Guirguis, M., Bestavros, A., and Matta, I. (2004b). Exploiting the transients of adaptation for RoQ attacks on internet resources. *In Proceedings of the 12th IEEE International Conference on Network Protocols (ICNP'04)*.

[Guirguis et al., 2005] Guirguis, M., Bestavros, A., Matta, I., and Zhang, Y. (2005). Reduction of quality (RoQ) attacks on internet end systems. *In Proceedings of INFOCOM,*.

[Guirguis et al., 2007] Guirguis, M., Bestavros, A., Matta, I., and Zhang, Y. (2007). Reduction of quality (RoQ) attacks on dynamic load balancers: Vulnerability assessment and design tradeoffs. *In Proceedings of INFOOM*.

[Hang et al., 2006] Hang, W., Zanji, W., and Jingbo, G. (2006). Performance of DSSS against repeater jamming. *Proceedings of the IEEE International Conferenece on Electronics, Circuits, and Systems (ICECS)*, pages 858–861.

[Huang and Li, 2001] Huang, X. and Li, Y. (2001). The multicode interleaved DSSS system for high speed wireless digital communications. *IEEE International Conference on Communications, 2001.*, pages 2990–2994.

[IEEE, 2003] IEEE (October 2003). Wireless medium access control(MAC) and physical layer(PHY) specifications for low-rate wireless personal area networks(LR-WPANs). *IEEE Standard, 802.15.4-2003*.

[Karlof and Wagner, 2003] Karlof, C. and Wagner, D. (2003). Secure routing in wireless sensor networks attacks and countermeasures. *In Proceedings of the 1st IEEE International Workshop on Sensor Network Protocols and Applications*, pages 113–127.

[Kuorilehto et al., 2009] Kuorilehto, M., Suhonen, J., and Hannikainen, M. (2009). Ultra-low energy wireless sensor networks in practice. *WILEY Press*, pages 133–142.

[Kuzmanovic and Knightly, 2006] Kuzmanovic, A. and Knightly, E. (2006). Low-Rate TCP-targeted denial of service attacks and counter strategies. *IEEE/ACM Trans. Netw.*, pages 683–696.

[Kyasanur and Vaidya, 2005] Kyasanur, P. and Vaidya, N. (2005). Selfish mac layer misbehavior in wireless networks. *IEEE Transaction on Mobile Computing*, pages 502–516.

[Law and Palaniswami, 2009] Law, Y. and Palaniswami, M. (2009). Energy-efficient link-layer jamming attacks against wireless sensor network mac protocols. *Transactions on Sensor Networks (TOSN)*, pages Vol. 5, No. 1, Article 6.

[Li et al., 2007] Li, M., Koutsopoulos, I., and Poovendran, R. (2007). Optimal jamming attacks and network defense policies in wireless sensor networks. *IEEE INFOCOM*, pages 1307–1315.

[Li et al., 2009] Li, X., Zhang, Z., Zhan, Y., and Sun, Z. (2009). Performance research and simulation on DSSS system with short spreading code against single frequency. *5th International Conference on Wireless Communications, Networking and Mobile Computing, 2009.*, pages 1–4.

[Lin and Noubir, 2004] Lin, G. and Noubir, G. (2004). On link-layer denial of service in data wireless lans. *Journal on Wireless Comm. and Mob. Computing*, pages 273–284.

[Liu et al., 2010] Liu, Y., Ning, P., Dai, H., and Lliu, A. (2010). Randomized differential dsss: Jamming-resistant wireless broadcast communication. *Proceedings of IEEE INFOCOM 2010*, pages 1–9.

[Mallik et al., 2000] Mallik, R., Scholtz, R., and Papavassilopoulos, G. (2000). Analysis of an on-off jamming situation as a dynamic game. *IEEE Transactions of Communications*, pages 1360–1373.

[McCune et al., 2005] McCune, J., Shi, E., perrig, A., and Reiter, M. K. (2005). Detection of denial-of-message attack on sensor network broadcasts. *In Proceedings of IEEE Symposium on Security and privacy*, pages 64–78.

[Negi and Perrig, 2003] Negi, R. and Perrig, A. (2003). Jamming analysis of mac protocols. *Carnegie Mellon Technical Memo*.

[Poisel, 2006a] Poisel, R. A. (2006a). Modern communication jamming principles and techniques. *Artech House Publishers*.

[Poisel, 2006b] Poisel, R. A. (2006b). Modern communication jamming princples and techniques. *Artech House Publishers*.

[Polastre et al., 2005] Polastre, J., Szewczyk, R., and Culler, D. (2005). Telos: Enableing ultra-low power wireless research. *Proc. Fourth Int'l Symp. Information Processing in Sensor Networks*, pages 13–24.

[Pöpper et al., 2009] Pöpper, C., Strasser, M., and Čapkun, S. (2009). Jamming-resistant broadcast communication without shared keys. *Proceedings of the 18th conference on USENIX security symposium*, pages 231–248.

[Seggern, 2006] Seggern, D. H. V. (2006). CRC standard curves and surfaces with mathematica, second edition (chapman & hall/crc applied mathematics and nonlinear science). *Chapman & Hall/CRC*.

[Severino et al.,] Severino, R., Jurcik, P., and Koubaa, A. http://www.open-zb.net/index.php.

[Shi and Perrig, 2002] Shi, E. and Perrig, A. (November 2002). Designing secure sensor networks. *Wireless Communications Magazine*, pages 38–43.

[Stallings, 1995] Stallings, W. (1995). Network and internetwork security: Principles and practice. *Prentice-Hall*.

[Strasser and Pöpper, 2008] Strasser, M. and Pöpper, C. (2008). Jamming-resistant key establishment using uncoordinated frequency hopping. *Proceedings of IEEE Symposium on Security and Privacy*, pages 64–78.

[Sun et al., 2007] Sun, H., Hsu, S., and Chen, C. (2007). Mobile jamming attack and its countermesure in wireless sensor networks. *International conference on advanced information networking and application workshop*.

[Technology,] Technology, C. Micaz mote data sheet. *http://www.xbow.com*.

[Xu et al., 2006] Xu, W., Ma, K., Trappe, W., and Zhang, Y. (2006). Jamming sensor networks: attack and defense strategies. *IEEE Network*, pages 41–47.

[Xu et al., 2005] Xu, W., Trappe, W., Zhang, Y., and Wood, T. (2005). The feasibility of launching and detecting jamming attacks in wireless networks. *In Proceedings of ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, pages 46–57.

[Ye et al., 2004] Ye, F., Luo, H., lu, S., and Zhang, L. (2004). Statistical en-route detection and filtering of injected false data in sensor networks. *Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies, INFOCOM 2004.*, pages 2446–2457.

[Yu and Xiao, 2006] Yu, B. and Xiao, B. (2006). Detecting selective forwarding attacks in wireless sensor networks. *In Proceedings of the 20th International Parallel and Distributed Processing Symposium (SSN2006 workshop)*, pages 1–8.

[Zhan et al., 2005] Zhan, Y., Cao, Z., and Lu, J. (2005). Spread-spectrum sequence estimation for DSSS signal in non-cooperative communication systems. *IEE Proceedings Communictions*, pages 476–480.

[Zhang and Kitsos, 2009] Zhang, Y. and Kitsos, P. (2009). Security in RFID and sensor network. *CRC Press*, pages 293–317.

[Zhang et al., 2005] Zhang, Y., Li, D., Chen, L., and Lu, X. (2005). Detection of Denial-of-Message Attacks on Sensor Network Broadcasts. *IEEE Symposium on Security and Privacy, 2005*, pages 64–78.

[Zhu et al., 2004] Zhu, S., Setia, S., Jajodia, S., and Ning, P. (2004). An interleaved hop-by-hop authentication scheme for filtering of injected false data in sensor networks. *Proceedings of IEEE Symposium on Security and Privacy*, pages 259–271.