

SECURITY FROM LOCATION

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF
AERONAUTICS AND ASTRONAUTICS
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Di Qiu

December 2009

Abstract

The emergence of the Internet and personal computers has led to an age of unprecedented information access. The proliferation of Internet connectivity, personal computers, and portable, high density data storage has put volumes of data at one's fingertips. While the spread of such technology has increased efficiency and knowledge, it has also made information theft easier and more damaging. One common expression of information theft is a data storage disk or equipment containing sensitive or valuable information. For example, the U.K. government lost computer disks that contain personal information on almost half of the country's population. The information includes names, addresses, insurance numbers, bank account details, etc.

These emerging problems have made the field of information security grow significantly in recent years. This thesis develops a new means to provide more protection against information loss, named geo-security or location-based security. This new technology is well suited to help mitigate the above described data loss scenario. Precise location and time information can be used to restrict access of the system or equipment at certain locations and time frames. This study bridges the two fields of navigation and security, and provides experimental support for the concept using location information for security.

This thesis designs a theoretical framework that provides a general analysis of geo-security, and quantifies the reliability and security of a geo-security system. The qualitative part of this analysis includes navigation signal definition, system design, performance standards, threat model and attack scenarios, and mitigation of the threats/attacks. The quantitative part of this analysis measures and quantifies location-dependent navigation parameters based on information theory, and evaluates

the consistency, spatial decorrelation, and entropy of these parameters for a variety of relevant navigation systems. Next, this thesis demonstrates geo-security using Long Range Navigation (Loran) and Wi-Fi as case studies. Experiments were conducted to evaluate the performance based on the designed framework. They illustrate the trade space between permitting access to authorized users and denying access to attackers. Finally, error-tolerant algorithms, named fuzzy extractors, are developed to improve the availability and reliability of location-based security systems given the constraints of real-world navigation systems and signal propagation characteristics.

Acknowledgments

This thesis would not have been accomplished without the kind support of many people whose contributions I gratefully acknowledge. First, I would like to thank my principal advisor, Professor Per Enge, for his support, guidance, and encouragement throughout my journey of the doctoral program. Not only he has provided technical direction throughout my research in geo-security, but he has also given me enough freedom and many opportunities to expand my critical thinking and knowledge base, present myself, and build self-confidence. Professor Enge is a friend, mentor, and role model for me.

I would also like to thank Professor Dan Boneh for his technical guidance on cryptography and security. He always encourages me to defend myself, and I enjoy all the valuable discussions with him. In addition, I thank professor Boneh for his generosity in chairing my defense. I would like to express my gratitude to the other members of my reading and defense committee, Professor David Powell, Dr. Sherman Lo, and Professor Steve Rock. I thank them all for their insights and comments which have improved the writing of this dissertation. Thanks to Dr. Lo; he has given his time and effort to help me throughout my study at Stanford. His door is always open and he is always willing to help out whenever I have questions.

I would like to thank Mitch Narins of the Federal Aviation Administration for his support of research in Loran, which made this work possible. Thanks also go to Dr. Ben Peterson, Captain, USCG (retired) and Lt. Kirk Montgomery, USCG (retired). It has been a great pleasure to work and learn from them. I am grateful to Dr. Greg Johnson, Ruslan Shalaev, and Christian Oates of Alion Science & Technology for providing us the Stanford Seasonal Monitor data collection equipment. In addition,

I thank Lt. Christopher Dufresne of the USCG Loran Support Unit and Jim Shima of Symmetricon for the use of their equipment in my testing.

I am grateful for the help and comments of my colleagues in the GPS laboratory. First, I would like to thank Dr. Sam Pullen, who advised me in my first research project at Stanford. He has always encouraged me and given me constructive advice on my research. Thanks also go to Dr. Todd Walter, Dr. Eric Phelts, Dr. Dave De Lorenzo, Dr. Juan Blanch, Dr. Lee Boyce, Doug Archdeacon, Godwin Zhang, Dr. Seebany Datta-Barua, Dr. Hiroyuki Konno, Ming Luo, Nick Alexeev, Tsungyu Chiou, Jiwon Seo, Sherann Ellsworth, Dana Parga.

Last but not least, I would like to thank my parents, Yaping Zhao and Yuhua Qiu. Throughout my life, I have been fortunate to receive guidance, inspiration, encouragement, and love from them. Their support has helped me make it to this point today. I dedicate this thesis to them.

Contents

Abstract	iv
Acknowledgments	vi
1 Introduction	1
1.1 Background of Cryptography and Security	2
1.1.1 Cryptography and Security through The Ages	2
1.1.2 Fundamentals of Cryptography and Security	5
1.2 Motivation	7
1.2.1 Security Threats in the Information Age	8
1.2.2 Design Considerations	12
1.3 Geo-Security	13
1.3.1 Prior Work	14
1.3.2 Geolock to Geotag: How is the Proposed Geo-security Different from Previous Work?	17
1.3.3 Geotag Applications	18
1.4 Contributions	20
1.5 Outline	22
2 Theoretical Framework	23
2.1 Geo-Security Basics	23
2.1.1 System Model	23
2.1.2 Geotag Generation	24
2.1.3 Geotag Matching	26

2.1.4	Physical Pseudo Random Function (PPRF)	28
2.2	Geo-Security Integrity	29
2.2.1	Performance Metrics	29
2.2.2	Spoofing Attacks	31
2.2.3	Defense against Spoofing	33
2.2.3.1	Tamper-Resistant Device	33
2.2.3.2	Signal Authentication Scheme - Timed Efficient Stream Loss-tolerant Authentication	34
2.2.4	Location-Based Attacks	37
2.2.4.1	Location Brute Force Attack	37
2.2.4.2	Selective Delay Attack	41
2.3	Continuity	43
2.4	Temporal and Spatial Entropy	46
2.4.1	Temporal Entropy to Measure Continuity	47
2.4.2	Spatial Entropy to Measure Integrity	47
2.4.3	Information Entropy to Bound Geotag Length	48
2.5	Conclusions	50
3	System Design for Loran	51
3.1	Loran Background	52
3.1.1	System Configuration	53
3.1.2	Properties Beneficial to Geo-Security Implementation	58
3.1.3	Loran Data Channel (LDC)	59
3.2	Integrity Analysis	63
3.2.1	TESLA Using New Loran Data Channel	63
3.2.2	Parameter Spatial Decorrelation	72
3.2.3	Resolution of Loran Geotag	77
3.2.4	Location-Based Attacks	80
3.3	Continuity Analysis	85
3.3.1	Seasonal Monitor Data	86
3.3.2	ASF Mitigation	88

3.3.3	Continuity Evaluation Using Seasonal Data	90
3.4	Loran Information Measure	92
3.4.1	Temporal Entropy	92
3.4.2	Spatial Entropy	93
3.4.3	Information Entropy to Bound Geotag Length	95
3.5	Conclusion	96
4	System Design for Wi-Fi	98
4.1	Wi-Fi Signal Characteristics	98
4.2	Continuity Analysis	104
4.3	Conclusion	107
5	Fuzzy Extractors to Reduce Continuity Risk	109
5.1	Continuity Risks	109
5.1.1	Error Model	110
5.1.2	Distance Measures	111
5.2	Fuzzy Extractor	112
5.2.1	Definitions	113
5.2.2	Euclidean Metric Fuzzy Extractor	114
5.2.3	Hamming Metric Fuzzy Extractors	117
5.2.4	Summary of the Proposed Fuzzy Extractors	121
5.3	Reproducibility Analysis	121
5.4	False Accept Rate for Security Analysis	126
5.5	Tradeoff Analysis	129
5.6	Conclusion	132
6	Conclusions	133
6.1	Results and Contributions	133
6.2	Directions for Future Work	136
A	Information Theory Review	138

B	Pattern Classification for Geotag Generation	140
B.1	Review of Classifiers	141
B.2	Classifier-Based Geotag Generation	143
B.2.1	Experimental Results	144
C	Acronyms and Symbols	150
	Bibliography	154

List of Tables

1.1	Cryptographic function roles	7
3.1	Loran station phase codes	54
5.1	Fuzzy extractor algorithms for performance comparisons	132

List of Figures

1.1	Egyptian hieroglyphic writing	3
1.2	Spartan skytale	4
1.3	Vigenere cipher	4
1.4	Enigma and its wiring diagram	5
1.5	Symmetric and public-key algorithms	6
1.6	Breaking news on U.K. government data loss	9
1.7	Movie piracy rate worldwide	10
1.8	CyberLocator achieves location-based authentication [Denning and Mac- Doran, 1996]	15
1.9	Geoencryption overview	16
1.10	Geo-security system demonstration	18
1.11	Geotag for Loopt	19
2.1	H_0 and H_1 hypothesis distributions	30
2.2	Tradeoff between P_{FR} and P_{FA}	31
2.3	Geo-security attack model	33
2.4	One-way key chain generation	35
2.5	TESLA setup	36
2.6	Location brute force attack	38
2.7	Spatial decorrelation of uncorrelated parameters	39
2.8	Selective delay attack	42
2.9	Illustration of spoofing attacks in action	42
2.10	Selective delay attack: upper bound on the number of trials	44
2.11	False reject rate as a function of quantization error e	46

2.12	A reduction in spatial entropy and total information with an increase in temporal entropy. The temporal entropy is derived by varying the quantization steps.	49
3.1	Signal opportunities for geo-security	53
3.2	Chapter 3 organization	54
3.3	Loran world coverage map	55
3.4	Loran stations over U.S. and Canada	55
3.5	Loran GRI timing diagram	56
3.6	Ideal Loran pulse with positive phase code	56
3.7	Skywave interference	57
3.8	Hyperbolic positioning	58
3.9	PPM 32 on the ninth-pulse	60
3.10	PPM 32 matched filter	61
3.11	32-state PPM probability of error	62
3.12	Message loss vs. packet loss rate	63
3.13	Loran transmitter [Picture source: US Coast Guard]	64
3.14	Stanford generated Key and MAC for TESLA demonstration	66
3.15	Circular TESLA chain on Middletown	66
3.16	Middletown signal field strength contour plot, signal strength in $\mu\text{V}/\text{m}$	68
3.17	Middletown authentication probability as a function of user locations	70
3.18	Authentication time is inversely proportional to data capacity and SNR.	71
3.19	Data collection setup and the test locations	72
3.20	Test points in a parking structure at Stanford University	73
3.21	Spatial decorrelation of Loran location-dependent parameters. Quantization steps were chosen based on the received SNR. (a) TD for different stations: $\Delta = 15$ m for George; $\Delta = 3$ m for Middletown; $\Delta = 15$ m for Searchlight. (b) Parameters from Middletown: $\Delta = 3$ m for TD; $\Delta = 150$ ns for ECD; $\Delta = 5$ dB for SNR.	74
3.22	Spatial decorrelation of TD: measurements and curve fitting	75
3.23	A process to choose an optimal quantization step	75

3.24	Spatial decorrelation comparison of quantization-based, kNN and SVM classifier-based geotag generation algorithms	76
3.25	Linear dimensionality reduction from two-dimensional to one-dimensional: examples of bad projection (left) and good projection (right)	77
3.26	Loran data collection setup: Locus H-field antenna, Locus SatMate 1030 receiver, and laptop for data logging.	78
3.27	Geotag 2-D visualization: parking structure (left); soccer field (middle); office building (right). The cells are created using Voronoi diagrams; each color represents a different geotag computed from measured location-dependent parameters.	79
3.28	Location-based attack data site	81
3.29	The parameter distance between the user and the attackers as a function of physical distance: (a) TD; (b) ECD; (c) Signal strength; (d) SNR.	82
3.30	N_{trials} for the location-based simple attack	83
3.31	The exhaustive search algorithm for the selective delay attack. \bar{v}_u is the user's discrete parameter vector, and \bar{v}_a is the attacker's discrete parameter vector.	83
3.32	N_{trials} for the selective delay attack	84
3.33	Stanford seasonal monitor station: a block diagram of connections and data collection equipment	87
3.34	Loran 90-day seasonal monitor data from the West Coast stations: TD, signal strength at peak, ECD and SNR.	88
3.35	Middletown TOA measurements and its histogram	89
3.36	TOA correction: "Previous day is today's correction" (top) and TD (bottom)	90
3.37	Reproducibility of a Loran geotag	91
3.38	Temporal entropy as a function of FRR	93
3.39	Spatial entropy: parking structure (upper left); soccer field (lower left); office building (right)	94
3.40	Geotag length upper bound	95

4.1	Integration of Loran and Wi-Fi for geo-security	99
4.2	Wi-Fi data collection setup	100
4.3	Downtown Menlo Park (top); Spatial decorrelation (bottom left); AP spatial distribution (bottom right)	101
4.4	Wi-Fi RSS measurements in a residential area	102
4.5	RSS as a function of distance between AP and receiver	103
4.6	Geotag visualization in an office building: eight APs (left); four APs (right)	104
4.7	Residential: availability histogram (left); availability as a function of RSS (right). Low RSS signals are easier to lose track.	105
4.8	Residential: Tradeoff between FRR and geotag resolution	106
4.9	Performance analysis in the office building	107
4.10	Loran and Wi-Fi integration	108
5.1	90-day Middletown TD measurements with quantization grids, $\Delta = 50$ m	111
5.2	Fuzzy extractor construction for geo-security	113
5.3	A demonstration of Euclidean metric fuzzy extractor using Loran mea- surements: calibration data (top); verification data, 30 days after (bot- tom).	116
5.4	Mapping matrix for an RS-based fuzzy extractor	119
5.5	Quantization scenarios: best (left); worst(right)	122
5.6	Euclidean metric FE performance improvement	123
5.7	RS decoder representation	124
5.8	Performance of RS-based fuzzy extractor	126
5.9	Performance of SS-based fuzzy extractor	127
5.10	Security performance of Euclidean metric fuzzy extractor	127
5.11	Trade spaces: FAR and FRR (left); FRR and entropy (middle); FRR and average cell diameter (right)	131

6.1	A robust geo-security system demonstration: 1) Multiple signals to improve spatial decorrelation and security strength of geotags; 2) Fuzzy extractors to tolerate temporal variations and reduce continuity risks, such that $T(t_1) = T(t_2)$	134
B.1	Three test locations in a parking structure at Stanford University . . .	145
B.2	Multi-class linear classifier trained by the Perceptron algorithm . . .	146
B.3	The decision boundaries of kNN, $k = 8$	147
B.4	Multi-class SVM classifier by OAO decomposition: kernel argument = 1 (left); kernel argument = 5 (right)	148
B.5	FRR of SVM classifier-based geotag	148

Chapter 1

Introduction

Information security that protects the confidentiality, integrity, and availability of information from unauthorized access has become a significant issue in today's world. Government agencies, private companies, and individuals rely on computer systems and the Internet to deal with confidential information about their employees, customers, products, research, and financial status. A security incident can have far reaching consequences and negative impacts on a government, company's or individual's public relations, customer confidence and revenue. This dilemma makes information security essential to privacy and an essential component in an effective business strategy. Section 1.1 of this chapter introduces the brief history of cryptography, followed by some examples of security threats in today's information age in Section 1.2. Section 1.3 defines geo-security, gives the literature review, and discusses the differences between the focus of this dissertation and previous work. The possible application of location-based security services and the specific technical challenges are described. The fulfillment of these challenges based on the proper design of location-based security systems is the subject of this dissertation. Section 1.4 summarizes the research contributions, and draws the separate threads from each chapter together to demonstrate how to build a robust geo-security system.

1.1 Background of Cryptography and Security

In the last few decades, the study of cryptography has gained applicability and efficiency. However, the history of cryptography can be traced back to 3500 B.C.. *Cryptography* is the science of disguising the meaning of a message; the word is derived from the Greek word “kryptos” (hidden) [47]. A *cipher* is a mathematical function used for cryptography. Its importance is obvious – it is used everywhere today: online shopping, secure money transfers, bank account management, cellular phones, broadcast of TV channels, emails, data management, and so forth. Our lives would be quite different without cryptography. This section discusses the evolution and fundamentals of cryptography and security.

1.1.1 Cryptography and Security through The Ages

3500 B.C. - 1500 A.D.

The Egyptian Hieroglyphics, shown in Figure 1.1, might represent the first known cryptography. Hieroglyphs are picture languages that were used for religious, historical, and economic purposes by the ancient Egyptians [8]. They could be written on papyrus, painted, or carved into stone, and most often were used to decorate temples and monuments and keep records of the king’s possessions.

In ancient Mesopotamia, the oldest encipherment is a piece of cuneiform tablet, which contained the formula for making pottery glaze, dating from 1500 B.C.. Cuneiform signs were used to encrypt the formula. Ancient Greeks invented the Spartan skytale illustrated in Figure 1.2, which was a wooden stick wrapped with narrow strips of papyrus, leather, or parchment [8]. The sender wrote the message along the stick; then the strip was removed and passed to the messenger. If the receiver had the same size stick, he would be able to read the message by rewinding the strip. Julius Caesar used simple substitution ciphers, one of the simplest and most widely known encryption techniques. Each letter of the plaintext is replaced by a letter shifted down a fixed number of positions. For example, a shift of three would move A to D, B to E, and so on.



Figure 1.1: Egyptian hieroglyphic writing

The first occurrence of *Cryptanalysis*, the science of breaking ciphertext, was among the Arabs around the 8th century. By the 15th century, during the Renaissance, cryptography became important due to political competition and religious revolutions, and cryptanalysis began to emerge in the West.

The Beginning of Modern Cryptography: 1500 - World War II

The Vigenere Cipher, illustrated in Figure 1.3, was invented in 1587 by Blaise de Vigenere and was thought for a long time to be unbreakable. The encipherer chooses a keyword and repeats it until it matches the length of the plaintext, then the Vigenère table is used to perform encryption and decryption. Mary Queen of Scots used mono-alphabetic substitution ciphers to communicate with fellow conspirators in an attempt to overthrow her cousin, Queen Elizabeth I of England [52]. In addition, the Great Cipher, developed by the Rossignols, was used by Louis XIV to encrypt his secret messages.

Cryptography has also played a significant role during wars. Samuel Morse created Morse code in 1845. Morse code, which is an early form of digital communication, represents letters, numbers and punctuation marks by means of a code signal sent intermittently. For example, letter A is represented as '.-', a short pulse and a long

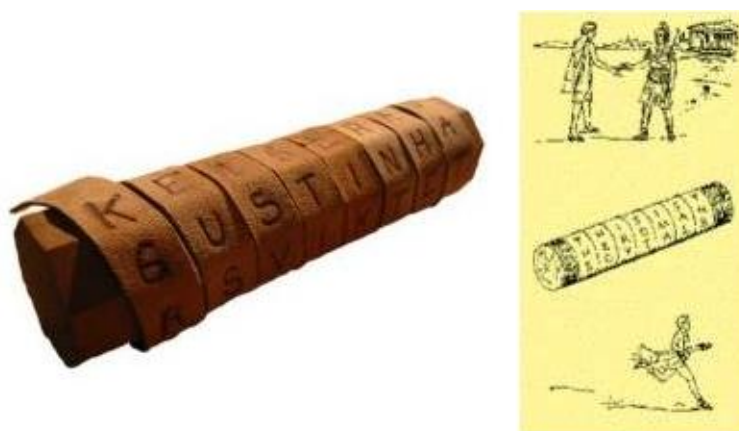


Figure 1.2: Spartan skytale

Plaintext	thiscryptosystemisnotsecure																																																						
Keyword	cipher																																																						
$A \leftrightarrow 0, B \leftrightarrow 1, \dots, Z \leftrightarrow 25$																																																							
<table><tr><td>19</td><td>7</td><td>8</td><td>18</td><td>2</td><td>17</td><td>24</td><td>15</td><td>19</td><td>14</td><td>18</td><td>24</td><td>18</td><td>19</td><td>4</td><td>12</td><td>8</td><td>18</td><td>13</td><td>14</td><td>19</td><td>18</td><td>4</td><td>2</td><td>20</td><td>17</td><td>4</td></tr><tr><td>2</td><td>8</td><td>15</td><td>7</td><td>4</td><td>17</td><td>2</td><td>8</td><td>15</td><td>7</td><td>4</td><td>17</td><td>2</td><td>8</td><td>15</td><td>7</td><td>4</td><td>17</td><td>2</td><td>8</td><td>15</td><td>7</td><td>4</td><td>17</td><td>2</td><td>8</td><td>15</td></tr></table> <hr/>		19	7	8	18	2	17	24	15	19	14	18	24	18	19	4	12	8	18	13	14	19	18	4	2	20	17	4	2	8	15	7	4	17	2	8	15	7	4	17	2	8	15	7	4	17	2	8	15	7	4	17	2	8	15
19	7	8	18	2	17	24	15	19	14	18	24	18	19	4	12	8	18	13	14	19	18	4	2	20	17	4																													
2	8	15	7	4	17	2	8	15	7	4	17	2	8	15	7	4	17	2	8	15	7	4	17	2	8	15																													
<table><tr><td>21</td><td>15</td><td>23</td><td>25</td><td>6</td><td>8</td><td>0</td><td>23</td><td>8</td><td>21</td><td>22</td><td>15</td><td>20</td><td>1</td><td>19</td><td>19</td><td>12</td><td>9</td><td>15</td><td>22</td><td>8</td><td>25</td><td>8</td><td>19</td><td>22</td><td>25</td><td>19</td></tr></table>		21	15	23	25	6	8	0	23	8	21	22	15	20	1	19	19	12	9	15	22	8	25	8	19	22	25	19																											
21	15	23	25	6	8	0	23	8	21	22	15	20	1	19	19	12	9	15	22	8	25	8	19	22	25	19																													
<div>↓</div>																																																							
Ciphertext	VPXZGIXIVWPUBTTMJPWIZITWZT																																																						

Figure 1.3: Vigenere cipher

pulse. The German ADFGVX cipher was first used by the German Army during World War I. The Enigma, depicted in Figure 1.4, is a device that allowed parties to communicate confidential documents without having to resort to clumsy and slow codebooks. Enigma was used most famously by Nazi Germany before and during WWII to encipher their communications. The German military was confident that the technology was unbreakable. A team of British code breakers along with some Polish mathematicians were able to crack the code and eavesdrop on German communications for years.

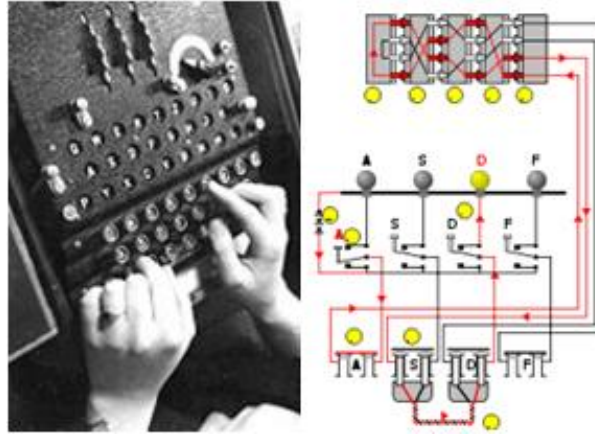


Figure 1.4: Enigma and its wiring diagram

Modern Cryptography and Security

The era of modern cryptography began with Claude Shannon, the father of mathematical cryptography. His work on information and communication theory provides a solid theoretical basis for cryptography and cryptanalysis. In addition, modern ciphers that rely heavily on the ideas of Shannon are Data Encryption Standard (DES), Advanced Encryption Standard (AES), public key cryptography, and Pretty Good Privacy (PGP), which are still in use today [52].

1.1.2 Fundamentals of Cryptography and Security

Although the history of cryptography is long and complex, its role in providing confidentiality, authentication, integrity, and nonrepudiation has not changed throughout the years. Confidentiality refers to the function that hides secret information from unauthorized entities. Authentication ascertains the origin of a message and ensures that a message cannot be faked by someone. Integrity verifies whether a message has been modified during a transit from a sender to a receiver. Nonrepudiation ensures that a sender cannot falsely deny the fact that he sent a message.

This section presents some basic cryptographic algorithms and functions [53]. To keep the secrecy of a message or plaintext, a key and a cipher for encryption can be used to convert the plaintext into a ciphertext, which is unreadable. A decryption

cipher, usually different from the encryption cipher, converts the ciphertext back into plaintext if the correct key is used. There are two types of general cryptographic algorithms: symmetric and asymmetric. The encryption key, also referred to as a session key or a secret key, can be calculated from the decryption key, and vice versa for symmetric algorithms. Asymmetric algorithms, often called *public-key algorithms*, are designed so that the encryption key differs from the decryption key. The encryption key can be made public so it is often called a *public key*: Anyone who has the public key can encrypt a message, but only a specific person with the corresponding decryption key can decrypt the message. Thus, the decryption key is referred to as a *private key*. Symmetric and public-key algorithms are illustrated in Figure 1.5. Symmetric algorithms are fast and efficient; on the other hand, asymmetric algorithms are slow. For instance, symmetric algorithms are at least 1000 times faster than public-key algorithms. Mathematically, it is computationally difficult to break public-key algorithms or deduce the private key from the public key. Therefore, in practice, a hybrid algorithm that combines the symmetric and public-key algorithms is often used; public-key algorithms are used to secure and distribute the session key used in a symmetric algorithm.

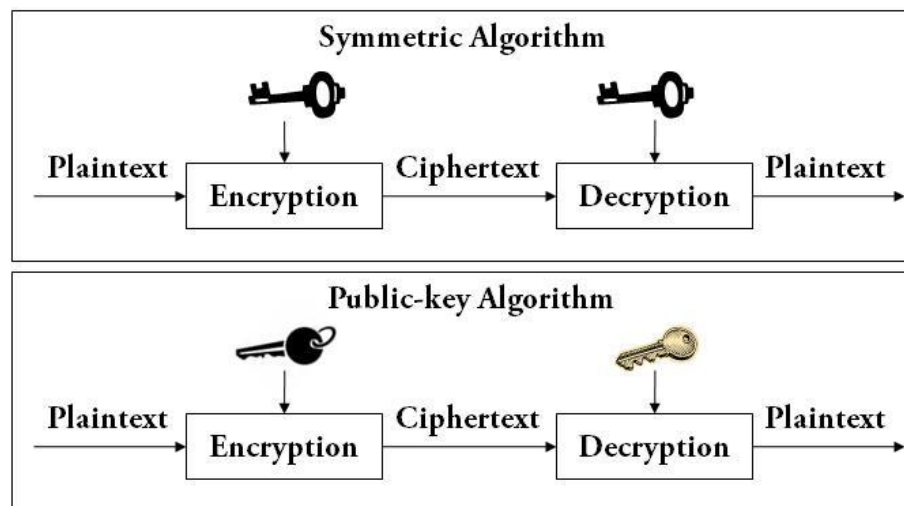


Figure 1.5: Symmetric and public-key algorithms

The one-way function, the hash function, and the message authentication code

(MAC) are fundamental building blocks in many cryptographic algorithms and protocols. One-way functions are easy to compute, but hard to invert. “Hard” means that it would take a reasonably significant amount of time and computation power to invert the function. The one-way hash function is another important component in many protocols. It converts variable-length inputs into a fixed-length output, called a hash value or *hash digest*. The one-way hash function has the property of one-wayness. In addition, it is collision resistant, such that it is computationally infeasible to generate the same hash values from two different inputs. Examples of one-way hash functions are MD5, SHA-1, and SHA-256. A MAC is a one-way hash function with the additional use of a secret key. A MAC protects the authenticity and integrity of a message by allowing verifiers to detect any changes to the message content. Although a hash function and a MAC work similarly in terms of data protection, they possess different security requirements. A MAC function is more robust to forgery under chosen-plaintext attack. This property means that it requires unfeasible amounts of computation for an attacker to guess the MAC of other messages even if the attacker can generate the MACs for the messages of his choosing. Whereas hash functions and MAC are symmetric, the common public-key based authentication schemes, such as RSA (stands for Rivest, Shamir, and Adleman who first publicly described it), digital signature algorithm (DSA), and discrete logarithm signature, are asymmetric. The roles of different cryptographic functions are displayed in Table 1.1.

Services	Encryption	Hash	MAC	Signature
Confidentiality	x			
Authentication		x	x	x
Integrity		x	x	x
Non-repudiation				x

Table 1.1: Cryptographic function roles

1.2 Motivation

In a generic cryptographic system, the possession of a key, signature, or verification tag is sufficient to establish user authenticity. According to Kerckhoffs’ principle [53], the

secrecy of a key provides security. The randomness and secrecy of a key can be derived from different information sources. The most commonly used means is “something you know,” such as a password or a PIN [56]. A metal key, an ID card, a , a car registration plate are the examples of “something you have.” “Something you do” refers to the patterns of users’ physical behaviors, such as handwriting, accent, keyboard strokes, habits. Finally, “something you are,” such as biometrics, indicates users’ physiological characteristics or *features*. Popular biometric features are fingerprints, voice, iris, retina, face, and palm geometry.

Nevertheless, there are inconveniences and weaknesses in the conventional cryptographic systems; thus, it is always helpful to provide an additional level of protection. “Something you know,” such as a password, is difficult to memorize. Even today, many users do not employ very strong passwords; thus, a simple dictionary attack can break weak passwords. Eight character passwords composed of a mixture of numbers and letters can be recovered within 60.5 hours on supercomputers that have a speed of 1 billion passwords per second [5]. “Something you have,” such as keys and ID cards, can be lost or stolen. Physical behavior, “something you do,” lacks accuracy and does not provide enough information to construct a key or a verification tag. Finally, the principal concern of biometric cryptosystems is the problem of privacy. Biometric systems use database technology to store users’ biometric features, which makes the privacy violation easier and more damaging. Many of these limitations of the conventional cryptographic algorithms can be ameliorated by the incorporation of better methods. Adding location information to existing encryption or authentication is called *geo-security*, which is the subject of this dissertation. This dissertation will show that geo-security has advantages including the fact that location-dependent signal characteristics cannot be lost or forgotten; they are extremely difficult to copy, share, and distribute; and they are easy to acquire and quantify.

1.2.1 Security Threats in the Information Age

In addition to the structural weaknesses in the conventional security systems discussed above, there are many other threats that pose a challenge to information security.

This section addresses these threats with examples.

“The Biggest Data Disaster”

In early November 2007, the U.K. government lost computer disks that contained the personal information on 25,000,000 of Britain’s 60,000,000 citizens [54]. This event is being called “the biggest data disaster” of the information age, illustrated in Figure 1.6. The computer disks were lost while being sent from one government agency to another by mail. The personal information on the lost disks contained intimate details including names, addresses, dates of birth. The true value of this data loss is difficult to predict, but one estimate of the price tag was \$500 million. Personal information is hot on the identity black market. For instance, if lost bank account details and credit card numbers fall into criminals’ hands, the problem is exacerbated, costing both banks and their clients.

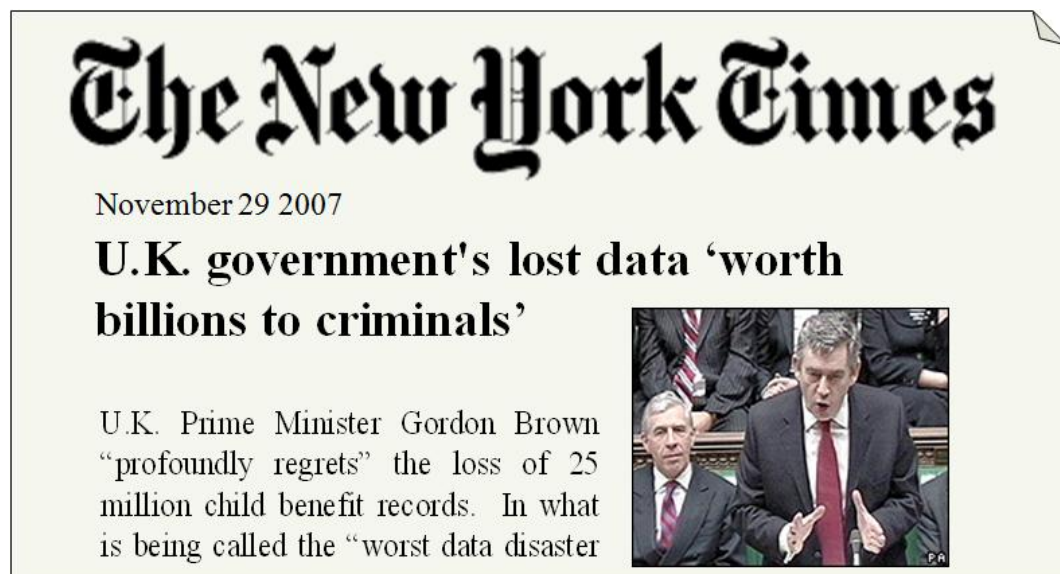


Figure 1.6: Breaking news on U.K. government data loss

Movie Piracy

Another common type of information theft is the unauthorized copying and distribution of copyrighted material. Today, one can obtain pirated versions of the latest

movies, often prior to their release, by making a quick visit to a file-sharing network or a less-than-reputable shop. Surprisingly, the source of the pirated material is often Hollywood insiders, such as the employees of the post-production shop, or individuals selected to receive pre-release screener DVDs [28]. The pirated copies of *X-Men Origins: Wolverine*, *The Incredibles*, *Terminator 3* or other major movies have appeared on the Internet days or weeks before their first theatrical release. According to the movie piracy assessment report, U.S. motion picture studios lost \$6.1 billion due to movie piracy in 2005. 62% of this loss originated from the piracy of hard goods such as DVDs and 38% due to Internet piracy [4]. Movie piracy has become a top economic issue for the movie industry; the problem is not only in the United States but also in many other countries. Figure 1.7 shows the percent of potential market lost to piracy in different countries. China, Russia and Thailand have the highest piracy rates worldwide.

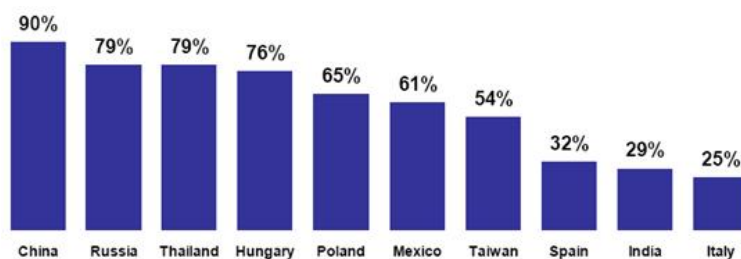


Figure 1.7: Movie piracy rate worldwide

Laptop Theft

Another common occurrence is the theft of equipment containing sensitive or valuable information. The theft and loss of laptops containing personal information, such as social security numbers, personal financial information, or credit card numbers, is becoming increasingly common. These thefts can occur in the most surprising places. For example, Qualcomm CEO Irwin Jacobs had his laptop stolen shortly after delivering a speech to a small group from the Society of American Business Editors. He had left it unattended for a few minutes to field questions from the audience [45]. The technology and information that make laptops so useful to users

also make them valuable prizes for thieves. Some laptop thieves head straight to the pawn shop after stealing the equipment, whereas others attempt to grab valuable proprietary information from the computer. In the case of Jacobs, while his hardware was estimated to be worth about \$4,000, the sensitive information on the disk was thought to be worth millions.

As a result, the emergence of the Internet and personal computers has led to an age of unprecedented information content and access. The proliferation of Internet connectivity, personal computers, and portable, high density data storage has put volumes of data at one's fingertips. While the spread of such technology has increased efficiency and knowledge, it has also made information loss and theft easier and more damaging.

More Security Issues

While the above incidents are examples of the external threats to information security, there are internal threats as well that raise concerns about how information is secured and maintained. First, a number of vulnerabilities can impact the risk of a security threat. In the field of cryptography and security, vulnerabilities include the weaknesses in security systems, policies, procedures, and implementations that can facilitate a successful exploitation of a threat. An attacker or adversary is a malicious entity whose aim is to discover data, destroy important information, spoof a device or system with false messages, or deny services. The following is a list of examples of common vulnerabilities.

- Poorly designed algorithms or protocols. If a security protocol is poorly designed, an attacker could easily learn the necessary information, thereby circumventing any security a system might have offered.
- Improper implementation of security protocols. Improperly implemented security protocols may introduce latency on machines and leave an attacker the opportunity to bypass a security check.
- Hardware failure, system, or communication disruptions. Loss of data or service may occur due to hardware failure or system disruptions.

- Utility failure (power, water, heat, etc.). This risk is related to physical security practices and can impact on the risk of a threat to any information security system.
- Improper handling of information. Poorly handled information is a leading cause behind critical vulnerabilities that exist in systems and applications.
- Malicious software. It is any software designed to disrupt service, that is, viruses, Trojans, worms, back doors. The damage can vary from partial to full control of one's computer without his ability to easily find out.

One of the greatest issues that threatens information security is that users in general have a low awareness of the risks. The majority of Internet and mobile device users do not know that there are spyware and malware out there that steal their data or important information. In addition, many use simple passwords to manage their data or electronic devices. As mentioned, short and simple passwords can be easily cracked using dictionary attacks. Surprisingly, the most commonly used password is the word “password” [24].

Attackers do not follow rules. They can attack a system using techniques the designers never thought of. In many security applications, attackers do not need even physical access to break or attack a system. An attacker in Russia or China can hack into the computer systems of the U.S. government or bank accounts in other countries. All of these security issues discussed above make the design and implementation of a security algorithm difficult.

1.2.2 Design Considerations

All these emerging problems have caused the field of information security to expand significantly in recent years. This dissertation discusses how to use location information as a means to secure physical access control to prevent unauthorized persons from accessing data or electronic devices. Location-based security is suitable for all of the above scenarios, viz., data security, digital film distribution, and laptop security.

This dissertation focuses primarily on the modeling, performance evaluation, and demonstration of location-based security or geo-security systems. Since the security is

derived from location information, it is very important to choose a robust navigation system to implement the protocol. The selected navigation system must be able to correctly detect and flag hazardous conditions, and be reliable in detecting problems. To define and achieve the performance specifications, this dissertation addresses the following fundamental questions to design a robust geo-security system.

1. What are the desired signal characteristics of radio frequency (RF) signals for implementing geo-security? How different is the performance with different navigation systems?
2. How secure is geo-security? A location-based security system must survive the following attack: an attacker owns a geo-security device and tries to make the device think it is somewhere else. A strong geo-security system does not allow the attacker to have a high success rate, that is, the *false acceptance rate* must be low. High attacker false acceptance rate weakens a geo-security system.
3. How reliable is geo-security? In other words, can the system function well most of time under all conditions? An instance of false rejection, in which the system fails to recognize an authorized user and rejects that person as an attacker, degrades the reliability of a geo-security system, and makes the system impractical. As a result, the *false rejection rate* must be low.

Precise answers to these questions would enable geo-security systems to strike the delicate balance between reliable operation and security.

1.3 Geo-Security

The term “geo-security” or “location-based security” refers to an algorithm that limits the access to information content or an electronic device to specified locations and/or times [41, 37, 38]. More generically, the restriction can be based on any set of location-dependent parameters.

Geo-security does not replace any of the conventional cryptographic means but instead increases security by augmenting current security systems. Even with very

strong passwords, geo-security provides extra security because it prevents authorized users from accessing sensitive data at insecure locations. For example, it is not appropriate for a client's bank official to be viewing your personal financial information at the local coffee shop, where strangers can walk by and pick off sensitive information, such as social security or account numbers. Users should only be able to access this data at trustworthy and secure locations.

In the scenario of laptop security discussed in Subsection 1.2.1, the valuable information on the laptop or hard drive can be encrypted so that it can be accessed only at the laptop owner's home or office. Needless to say, the highly sensitive data on many people's computers, such as the CEO of Qualcomm, could be greatly abused if it fell into the wrong hands. The theft of such data has the potential to jeopardize personal and national security. The prevention of such loss is an important reason for having geo-security.

1.3.1 Prior Work

In its basic form, location-based security can be used to ensure that data cannot be read or used outside a particular facility. Any attempts to access the secure information at an unauthorized location will result in location validation failure.

Denning and MacDoran developed location-based authentication in 1996 [16], which uses physical location such as latitude, longitude, and height to restrict the Internet access for remote users. To gain access to a host server, remote users derive the location signatures from their physical locations and the observations of all satellites in view, which are obtained from a GPS receiver. The location signatures are then configured into data packets and transferred to the host. The host processes its own simultaneously acquired satellite signals and verifies the users' locations to within an acceptable threshold, which is a design parameter (see Figure 1.8).

In 2003, Scott and Denning proposed geoencryption [17] for digital film distribution, that is, digital movie files can only be decrypted and exhibited inside an authorized movie theater. A brief overview of the system is demonstrated in Figure 1.9. Under this system, a content provider ("sender") distributes the encrypted

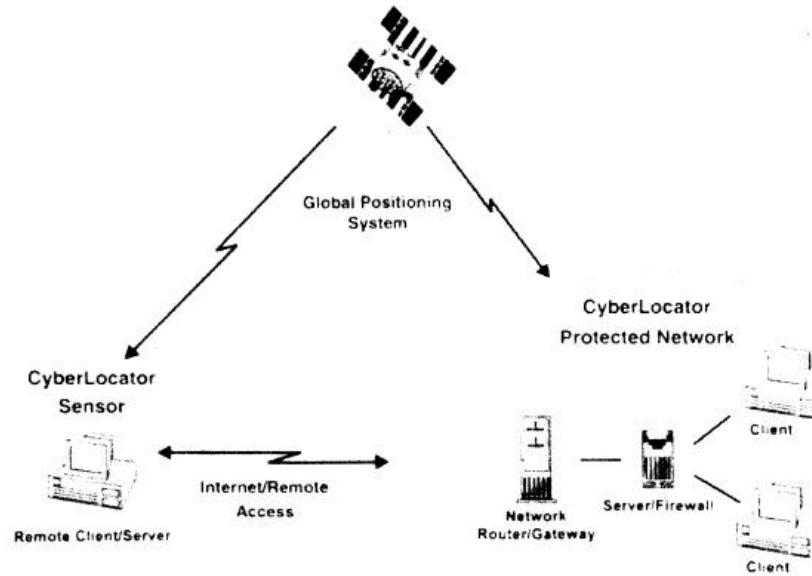


Figure 1.8: CyberLocator achieves location-based authentication [Denning and MacDoran, 1996]

film (ciphertext) to an authorized user (“recipient”). This transmission is done via many methods (such as satellite data links) and, as such, may be readily available to unauthorized users. The objective is to have films encrypted using the geoencryption protocol that is decryptable only at a specified location; the decryption process should fail and not reveal information about the plaintext if there is an attempt to decrypt the data at another location. Therefore, the geoencryption algorithm can be used to ensure that the film cannot be retrieved, except at the theater by authorized personnel.

Traditional encryption is an integral part of the system. The sender encrypts the data file or plaintext using a conventional cipher Advanced Encryption Standard (AES) with a random key. A *geolock* is derived from specific user location- (and time-) dependent parameters and generated by mapping the recipient’s physical location into binary bits. The geolock encrypted key is then encrypted again with a public-key cipher, such as RSA. To ensure authenticity of the sender/receiver, both the public key and the private key are distributed by a trusted third party, Certification Authority (CA). In order to enable a geoencryption system, a recipient should

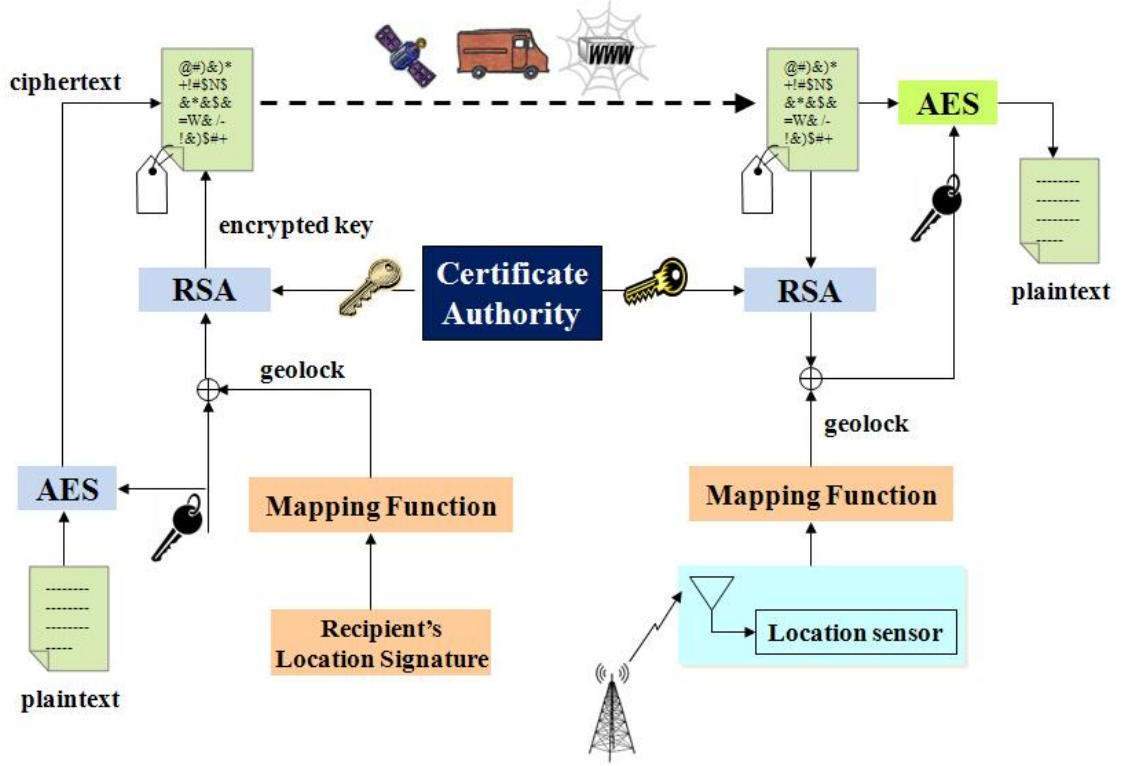


Figure 1.9: Geoencryption overview

have three channels to receive information. First, a data receiver is needed to capture a digital-encrypted data file. Furthermore, a navigation receiver is necessary to receive radio frequency (RF) signals, whose location-dependent parameters generate the geolock. A third channel is required for secure key exchange. Of course, there is a security risk if the key distribution channel is vulnerable or can be attacked. Once the geolock is computed from received navigation signals, it is used to decrypt the digital movie file. If the location verification is bypassed, the decryption process is performed successfully using the correct random key to obtain the digital movie file.

While Scott and Denning's system maps a user's physical location into a geolock for data encryption, this dissertation develops a location-dependent verification tag, which supports a wide range of applications. The difference between the proposed geo-security system and the above prior work is discussed in detail in the following subsection.

1.3.2 Geolock to Geotag: How is the Proposed Geo-security Different from Previous Work?

Geo-security in this dissertation uses a location verification tag, called a *geotag*, which extends the use of location-dependent information and ensures a user's physical location. Figure 1.10 illustrates the demonstration of a geo-security system with geotags. The system works in two steps: calibration and verification. A database of geotags is built at the calibration step; the verification step performs a matching algorithm and compares the derived geotag with the ones stored in the database, etc. A geotag is derived from received location-dependent signal characteristics instead of a user's physical location, such as latitude, longitude, and altitude, proposed by Denning and Scott. The examples of received location-dependent signal characteristics include signal propagation range, time-of-arrival, and signal strength. These location-dependent signal characteristics provide more *information entropy*; thus, they are more unpredictable. The information entropy [14] is a measure of uncertainty or randomness associated with a random variable, and quantifies the location information content in a geotag in this dissertation. Unpredictability produces randomness in a security system. In practice, it is difficult to keep a user's physical location secret. For instance, a movie theater's location is known to the public; as a result, a geolock derived from latitude, longitude, and altitude has little or no secrecy and cannot provide sufficient security. On the other hand, it is difficult to anticipate the location-dependent signal characteristics associated with users although their physical locations are known. Therefore, the proposed geotag in this dissertation computed from location-dependent signal characteristics provides more security than physical locations.

In addition, it is important to ensure that each linkage of a geo-security system is secure, including the signal itself. GPS signals are extremely weak since the satellites are 20,000 km away from the Earth [32]. Because of long propagation distances, GPS signals are very vulnerable to jamming and spoofing [49, 59, 34]. The objective of jamming is denial of service by masking the signal and injecting random noise. The objective of spoofing is to convince a target receiver that he is somewhere he

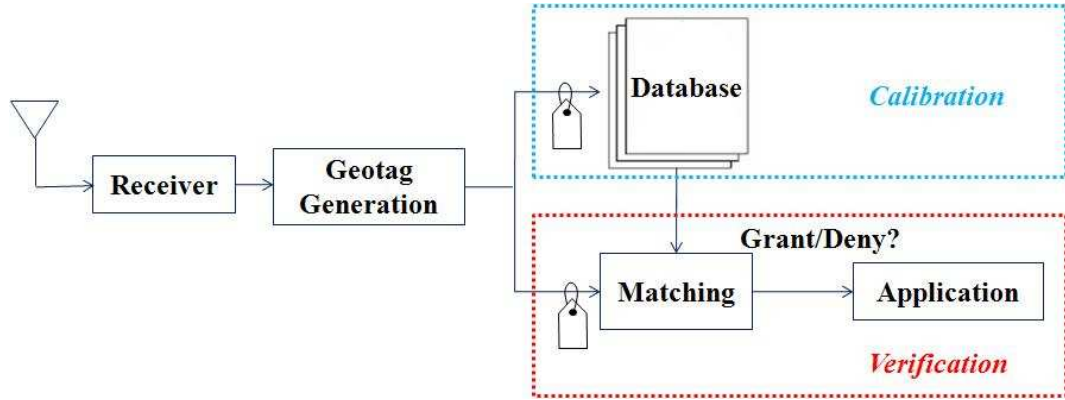


Figure 1.10: Geo-security system demonstration

is not. Both of these processes degrade the navigation performance, thus reducing the security level of geo-security systems. As a result, the geotag derived from GPS signals can be defeated by denial-of-service and spoofing attacks. This dissertation uses signals that are robust to jamming to implement geo-security and proposes methodologies to add the authentication feature to existing signals to protect against spoofing.

Since there is no well-developed theoretical framework to quantify the reliability and security of a geo-security system, one goal of this dissertation is to model a standard process to evaluate the system performance. Furthermore, this dissertation studies and compares the various signal characteristics, selects the signals that are suitable to implement geo-security, and validates geo-security performance using live signals by constructing a demonstration testbed.

1.3.3 Geotag Applications

Geotags help support a range of applications, such as tracking, health care, patient monitoring, emergencies, advertisement, marketing, and security services. This dissertation focuses on using geotags for two types of security applications: block-listing and white-listing.

An example of a block-listing application is the digital manners policy (DMP).

Technologies for DMP [20] attempt to enforce manners at public locations. A DMP-enabled cell phone can be programmed by the phone provider to turn off the camera inside a hospital, a locker room, or a classified installation. Or the phone can be programmed to switch to the vibrate mode inside a movie theater. Though some of these ideas may be highly controversial [48], this dissertation focuses only on the technical aspect of the application. The device downloads an updated list periodically. When the device encounters a geotag on this blocklist, it turns the camera off. When the device leaves the blocked location, the camera is turned back on. Hence, digital manners are enforced without ever telling the device the precise location.

Location-based access control is a white-listing example. Consider a location-aware disk drive: the drive can be programmed to operate only while safely in the data center. An attacker who steals the device will not be able to interact with it. Geocryption studied by Scott and Denning [51, 50] is a type of location-based access control.

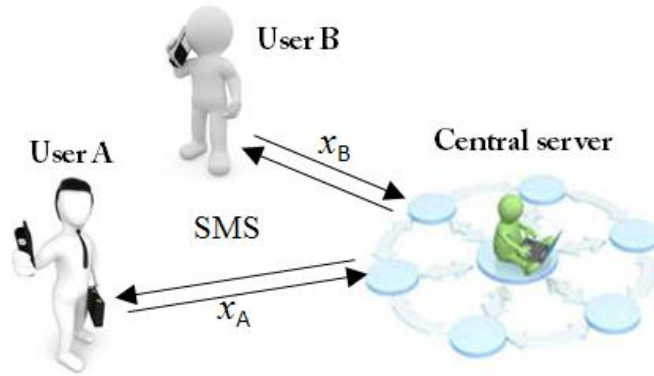


Figure 1.11: Geotag for Loopt

Another white-listing application is Loopt, which provides geosocial networking services to users, enabling them to locate friends via their GPS-based cell phones. To implement Loopt, a central server is required to compute geotags, perform matching algorithms, and notify users with SMS messages if they and their friends are in a given location. A diagram that illustrates Loopt implementation is shown in Figure 1.11. This dissertation proposes geotags, which do not reveal users' location information, for implementing Loopt to protect users' privacy.

1.4 Contributions

The objective of this work is to design and develop a robust geo-security system with appropriate implementation. To accomplish this goal, the research investigates the characteristics of navigation signals and examines the robustness of extracted location parameters for security applications. Several lines of investigation and performance evaluations emerge from the goal and constitute the bulk of this work. The specific contributions presented in this work are as follows:

Theoretical framework. The theoretical framework serves as the basis for conducting this research, and provides a direction to the study. A key challenge for a geo-security system and any system of this type is that it must meet strict performance requirements. Specifically, performance standards are imposed on two qualities of interest: integrity and continuity, which are quantified by false acceptance rate and false rejection rate in this dissertation. To evaluate the system performance, deciding whether a location is valid or not can be seen as a hypothesis-testing problem. In addition, this work plays a role in bridging the very different languages used by navigation and security fields. Furthermore, this theoretical framework teaches the radio signals with special characteristics: low temporal variation, and high spatial variation. These sought after properties will be discovered and used in the Loran and Wi-Fi demonstrations described below.

Demonstration: System validation using Loran. LOnG RAnge Navigation (Loran) is a navigation system developed during World War II that has played an important role in maritime navigation. This work demonstrates that Loran has many properties that are beneficial to the implementation of security applications. A signal authentication scheme is proposed on Loran to ascertain the origin of incoming signals and to preclude spoofing. The authentication performance is tested by implementing Stanford-generated authentication messages on a real Loran transmitter. The characteristics of the Loran signal are investigated by conducting various experiments. Seasonal monitor data are used to

quantify the consistency of Loran location-dependent parameters, while spatial data collected at different sites are analyzed to learn the spatial decorrelation of these parameters. Furthermore, a method to estimate the information entropy of each parameter is developed as the information entropy provides an upper bound on the geotag length, which can be a metric to bound the security level of a system.

Demonstration: System validation using Wi-Fi. Wireless fidelity (Wi-Fi) is a type of wireless local area network (WLAN) that uses the specifications in the IEEE802.11 standards. Although Wi-Fi was initially designed for communications between electronic devices, the proliferation of Wi-Fi has a growing interest in its use for indoor location-based applications today. A Wi-Fi device is installed on many personal computers, mobile phones, personal digital assistants (PDAs), and laptops. With growing acceptance in businesses, agencies, schools, and homes, many airports, hotels, coffee shops, and fast-food facilities offer public access to Wi-Fi networks. Thus, with its dense coverage in metro areas, Wi-Fi can be used to complement Loran indoors. As a result, the integrated system improves the spatial decorrelation, resolution, and information entropy in a derived geotag. The experimental results show that a resolution of 3-6 m can be achieved by integrating Loran and Wi-Fi in indoor environments.

Error tolerant algorithms to improve continuity. The signal characteristics should be consistent enough so that when a user is ready to verify, measurements at the same location will yield the same previously-generated geotag. In practice, temporal variation, which reflects instability or the degree of scatter within a particular parameter at a given location, increases the likelihood of mismatching geotags. Thus, fuzzy extractors, error-tolerant algorithms first designed for biometrics, are developed to reliably extract location-based signal characteristics from noisy location data. Three different fuzzy extractors are constructed for two location error types: the Euclidean metric and the Hamming metric. Experiments are conducted to demonstrate an improved continuity with fuzzy extractors.

1.5 Outline

This dissertation is organized essentially in the order of the contributions listed above. Chapter 2 describes the system model of a geo-security system. A theoretical framework is developed and used as a standard process to provide a general analysis and to quantify the continuity and integrity of a geo-security system. The framework includes 1) the desired navigation signal properties for security, 2) performance standards to evaluate the system, and 3) a threat model that defines the possible attacks to the system and mitigation of the attacks. The corresponding analysis measures and quantifies location-dependent parameters based on information theory and evaluates the consistency, spatial decorrelation and entropy of various parameters. The tradeoff between continuity and integrity is quantified using a receiver operating curve.

Chapter 3 demonstrates and validates geo-security using Loran as a case study. The background of Loran is first discussed. A demonstration testbed is built to analyze the continuity and integrity (reliability and security level) of a Loran geotag using various experiments. The demonstration results and analyses are also presented.

Chapter 4 extends the analysis and performance evaluation using Wi-Fi as a second case study. The signal characteristics of Wi-Fi are investigated for location-based services, and the improved integrity is demonstrated by integrating Loran and Wi-Fi. Chapter 5 describes the construction of three different fuzzy extractors, which reduce the system continuity risks caused by seasonal variations of location data. The fuzzy extractors are validated using Loran and Wi-Fi data, and the continuity improvement with different fuzzy extractors is presented.

Chapter 6 summarizes the results and contributions, and offers some thoughts about prospective directions for future research.

Appendix A presents a brief overview of information theory, which serves as the mathematical background to measure location-dependent parameters. Appendix B provides background information on pattern classification, which is a concept applied to generate robust geotags. The fields of security and navigation are laden with abbreviations and acronyms; a list of terms that are used throughout this dissertation are explained in Appendix C.

Chapter 2

Theoretical Framework

This chapter focuses on a theoretical framework for a geo-security system, that is, the methodologies for evaluating system performance. The theoretical framework of the study presents the theory which explains why the problem under study exists. Thus, the theoretical framework serves as a basis for subsequent chapters.

The discussion of theoretical framework is divided into four sections: 1) system modeling, 2) integrity analysis, 3) continuity analysis, and 4) location information measure. This chapter is organized as follows. In Section 2.1 the fundamental functions in a geo-security system, i.e., geotag generation and matching algorithm, are introduced. Section 2.2 provides detailed discussion on system integrity, which bounds the security of a geo-security system to a high degree of confidence. Continuity, which measures the reproducibility (i.e. repeatability) of a geotag, is presented in Section 2.3. Section 2.4 discusses a measure of location-dependent information to provide an upper bound on the derived geotag length.

2.1 Geo-Security Basics

2.1.1 System Model

A geo-security system proposed in this dissertation operates in two steps: calibration and verification, as mentioned in Chapter 1. The calibration phase builds a geotag

database for service areas: $\mathfrak{S} = \{T(\ell, t), \forall \ell \in \mathcal{L}\}$, where T is the geotag at the calibration associated with location ℓ , and t represents the time interval during which the geotag is generated, and \mathcal{L} represents the service areas. The use of time information for the geotag is optional. This requires that someone surveys the service areas with a location sensor, such as a Loran receiver or a Wi-Fi device, with an integrated geotag generation module. Geotags associated with the calibrated areas are computed based on the recorded location information and stored in a database for future use. At the verification phase, a user derives a geotag, $T'(\ell', t') \in \mathfrak{S}$, such that $\ell' \in \mathcal{L}$ using the same geotag generation device and matches it with the pre-computed ones in the database. If the two tags match, the user's location is validated, and the authorization of an application is granted; otherwise, the authorization is denied.

2.1.2 Geotag Generation

In this section, three geotag generation methods are introduced: a deterministic approach, a quantization-based approach, and a classifier-based approach. The methods differ in geotag representation, efficiency of computation, and implementation in practice.

Let $x = f(s(\ell, t))$ be the location-dependent parameters, where $s(\bullet)$ denotes the signals received at location ℓ and time t , and $f(\bullet)$ is the function performed in a receiver. Typical functions in a receiver include signal conditioning, digitizing, and parameter extraction. The extracted x is a vector, $x = [x_1, x_2, \dots, x_n]^T \in \mathbb{R}^{n \times 1}$, where n is the number of location-dependent parameters.

The deterministic approach simply utilizes the location-dependent parameter vector as a geotag, shown in Equation (2.1). This technique is similar to location fingerprinting [7] except a geotag is computed from various location-dependent parameters rather than solely the received signal strength.

$$T = x \in \mathbb{R}^{n \times 1} \quad (2.1)$$

The quantization-based tag generation algorithm consists of three steps: a receiver function, $f(\bullet)$, to extract location-dependent parameters from the received signals,

$s(\ell, t)$, a quantizer, $\mathcal{E}(\bullet)$, to quantize the parameters with adequate step sizes, $\Delta(\ell)$, and a mapping function, $\hbar(\bullet)$, to convert the quantized parameters into a binary string, T . The binary mapping process can be done using a hash function, which is one-way and collision resistant. A one-way hash function is a fundamental building block in many cryptographic algorithms and protocols [47], and outputs a fixed-length hash value regardless of the length of inputs. One-way-ness means that it is easy to compute but hard or computationally infeasible to invert the function. In addition, since it is collision resistant, it is difficult to generate the same hash values from two different inputs. Let q be the quantized parameter vector; its calculation is illustrated in Equation (2.2). All these vectors x , q , and Δ have the size n . The quantization steps can be determined based on the standard deviations of location-dependent parameters to tolerate a certain degree of nominal variation. The calculation of quantization steps will be discussed in Section 2.3.

$$\begin{aligned} q_i = \mathcal{E}(x_i) = k; \quad x_i \in S_k = [k\Delta_i, (k+1)\Delta_i) \\ k = 1, \dots, N, \end{aligned} \tag{2.2}$$

where S is the partition set, and N indicates the quantization levels corresponding to a particular Δ . Thus, the binary geotag can be calculated as

$$T = \hbar(q) \in \mathbb{Z}^{m \times 1}, \tag{2.3}$$

where m is the size of the hash value.

The third tag generation algorithm is developed based on pattern classification, which is the concept of assigning a physical object or measured data to one of the pre-specified groups, called *classes*, using *a priori* knowledge or statistical information. The patterns are the evaluated final decision from *classifiers* and represent the characteristics of location-dependent parameters. Mathematical models are used as the theoretical basis for the classifier design. In classification, a pattern is referred to as a pair of variables $\{x, \omega\}$, where x is a collection of location-dependent parameters and ω is the concept associated with the parameters, also called *class label*. This

dissertation selects three classifiers—linear discriminant analysis (LDA), k-nearest neighbor (kNN), and support vector machines (SVM)—to implement and generate a geotag. A detailed review on pattern classification is introduced in Appendix B. The classifier-based geotag generation algorithm consists of four steps: the same receiver function, $f(\bullet)$, to extract location-dependent parameters, a dimensionality reduction function to reduce high-dimensional data into low-dimensional, a model determined from the selected classifier and location data, and a mapping function $h(\bullet)$ to convert the class labels derived from the model into a binary string, T . Similar to the quantization-based approach, the classifier-based geotag is

$$T = h(\omega) \in \mathbb{Z}^{m \times 1}. \quad (2.4)$$

2.1.3 Geotag Matching

Different matching algorithms for the different geotag generation functions are described accordingly. Two matching algorithms – the nearest neighbor method (NNM) and a probabilistic approach – can be applied to the deterministic geotag. Let \mathcal{M} denote the matching function.

NNM is a common technique [43] used for indoor location estimation and pattern matching. The algorithm measures the distance between the location parameter vector from verification phase, T' , and the previously stored vectors in the database, \mathfrak{S} . The generalized distance measure, D , is defined in Equation (2.5), where w is a weighting factor and p is the norm parameter. For instance, $w = 1$ and $p = 2$ represent the Euclidean distance. Based on the calculated distances between T' and the previously computed $T \in \mathfrak{S}$, the geotag that produces the minimum distance is chosen. It is necessary to set an upper bound, d_0 , to guarantee that the location is registered at the calibration phase. A modification to NNM [46] that uses the standard deviation σ of the location parameters is named weighted nearest neighbor method (WNNM). The new distance measure is shown in Equation (2.6), where C is a covariance matrix, $C = E\{(x - \bar{x})^2\}$, and \bar{x} is the mean value of the location-dependent parameters. The matching function for a deterministic geotag is illustrated

in Equation (2.7), where \tilde{T} is the geotag associated with the authorized location.

$$D(x, x') = \frac{1}{n} \left(\sum_{i=1}^n \frac{1}{w_i} |x'_i - x_i|^p \right)^{\frac{1}{p}} \quad (2.5)$$

$$D(x, x') = [(x - x')^T C^{-1} (x - x')]^{\frac{1}{2}} \quad (2.6)$$

$$\mathcal{M}(\tilde{T}, T') = \begin{cases} 1 & \text{if } \arg \min_{T \in \mathfrak{S}} D(T, T') = \tilde{T}, D(T, T') \leq d_0; \\ 0 & \text{otherwise.} \end{cases} \quad (2.7)$$

The probabilistic approach models the geotag with conditional probability and uses a Bayesian concept to estimate location [43]. The location-dependent parameters and the standard deviations are estimated during the calibration phase. Assuming that the location-dependent parameters have Gaussian distributions, the probability density function shown in Equation (2.8) is used to compare the calculated likelihoods. The geotag that produces the maximum probability is chosen. The corresponding matching function is

$$P = \frac{1}{n} \sum_{i=1}^n \left[\frac{1}{\sqrt{2\pi}\sigma_i} \exp\left(-\frac{(x'_i - x_i)^2}{2\sigma_i^2}\right) \right], \quad (2.8)$$

$$\mathcal{M}(\tilde{T}, T) = \begin{cases} 1 & \text{if } \arg \max_{T \in \mathfrak{S}} P = \tilde{T}; \\ 0 & \text{otherwise.} \end{cases} \quad (2.9)$$

The matching process for the quantization-based and classifier-based geotag involves the correlation of T' and the previously stored geotags. The correlation function is

$$\mathcal{M}(\tilde{T}, T') = \begin{cases} 1 & \text{if } \frac{1}{m} \sum_{i=1}^m \tilde{T}(i) \oplus T'(i) = 1, \forall \tilde{T} \in \mathfrak{S}; \\ 0 & \text{otherwise.} \end{cases} \quad (2.10)$$

Other geotag generation techniques and matching algorithms could be applied to various signals and applications to perhaps achieve better performance. The quantization-based geotag and its corresponding matching algorithm are used for

the performance analysis in the remainder of this dissertation.

2.1.4 Physical Pseudo Random Function (PPRF)

By definition [47], a pseudo random function (PRF) is a deterministic function, $f : X \rightarrow Y$, which is efficient and computable. It takes two inputs $x, k \in X$. Consider x to be a variable, k to be a random seed, $f(x, k) = y$ and $y \in Y$.

This section shows that the interaction between RF signals and a receiver is a physical pseudo random function (PPRF). The inputs defined as the RF signals from multiple transmitters, which are a form of representation of a particular location. The deterministic function is a physical process to capture and condition the incoming signals, extract the location-dependent parameters, and map them into a geotag, which is the output of the PPRF. The random seed can be any randomness in the hardware devices such as antenna and receiver used to complete the physical process.

Some important properties of the derived PPRF are efficiency, distinguishability, and unpredictability. The physical process that converts RF signals to a geotag is efficiently computable. The second desired feature of location-based PPRF is the distinguishability. The algorithm must be able to generate distinguishable location tags given different input signals. In addition, the derived PPRF is unpredictable at a distance: someone who is twenty meters away from a target location cannot predict the geotag at the target. The experimental evidence for this claim will be discussed in Chapter 3.

There are some requirements on the physical system used to generate a geotag. First, the system should be easy to fabricate. This is important because a mass production of the system to be deployed is anticipated in the real world. In addition, the system should be structurally stable. It is expected that the derived geotag is reproducible; this requires not only the RF signals but also the physical system to remain stable over time.

2.2 Geo-Security Integrity

This section presents a broader and more practical view of geo-security system threats, places them in the context of a risk-based system, and outlines defenses. It is helpful to develop an attack model including all of the potential attacks or any attempt to break the system originating from the system weaknesses in both design and implementation. Security breaches are considered to be an integrity problem. Integrity is the ability to bound the geotag mismatched errors to a high degree of confidence. This means bounding the location-dependent parameter errors and variations for both authorized users and unauthorized users. Specifically, integrity is categorized into authenticity, the ability to protect against spoofing attacks, and security, the ability to resist any on-site attacks. This section analyzes and quantifies both spoofing and on-site attacks.

2.2.1 Performance Metrics

The performance metrics to quantify the geo-security attacks are discussed in this section. The problem of deciding whether the computed geotag is authentic is viewed as a hypothesis-testing problem. The following hypotheses are defined:

H_0 : Accept as an authorized user, $p \sim N(\mu_0, \sigma_0^2)$;

H_1 : Reject as an unauthorized user or attacker, $p \sim N(\mu_1, \sigma_1^2)$.

In cryptography, an attacker or adversary is a malicious entity whose aim is to discover secret data or access the system. Generally, attackers have some degree of technical skill. The task is to decide which of the two hypotheses, H_0 or H_1 , is true for an observed location measurement. The system can make two types of errors: 1) mistaking the measurements from the same location to originate from two different locations and accepting hypothesis H_1 when H_0 is true, called a *false reject*; and 2) mistaking the measurements from two different locations to originate from the same location and accepting H_0 when H_1 is true, called a *false accept*. The probability of a false reject and the probability of a false accept are given by

$$P_{FR} = P(H_1|H_0)P(H_0) \quad (2.11)$$

$$P_{FA} = P(H_0|H_1)P(H_1), \quad (2.12)$$

where H_0 and H_1 are *a priori* statistical descriptions of user and attacker locations.

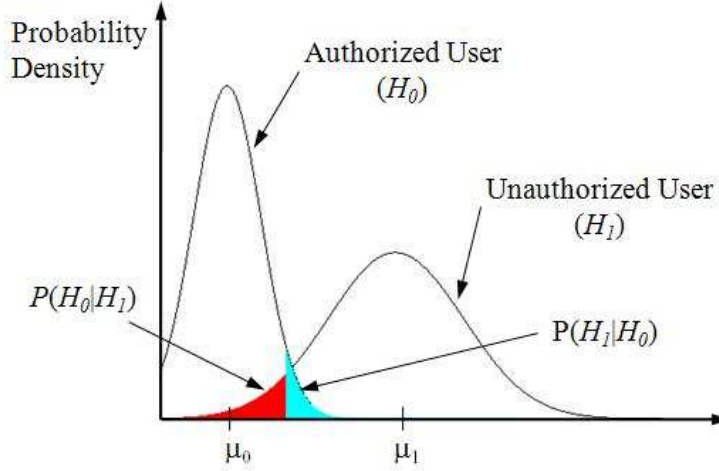


Figure 2.1: H_0 and H_1 hypothesis distributions

Figure 2.1 assumes that the location parameters have Gaussian distribution, which is not always true in practice. The Chebyshev bound can be applied in the case that the parameters are non-Gaussian. In probability theory, Chebyshev's inequality [25] characterizes the dispersion of data away from its mean value and puts an upper bound on the probability. It requires only two minimal conditions: (1) that the underlying distribution has a mean and (2) that the deviations away from this mean cannot be infinite. Chebyshev's inequality states that the probability that an observation is more than a standard deviations from the mean is at most $\frac{1}{a^2}$. Suppose $x \in \mathfrak{R}$, then Chebyshev's inequality is

$$P(|x - Ex| \leq a) \leq \frac{1}{a^2} \text{cov}(x), \quad (2.13)$$

where Ex and $\text{cov}(x)$ are the mean and variance of x .

Both the false reject rate (FRR), P_{FR} , and the false accept rate (FAR), P_{FA} , depend on the variations of the location parameters, the quality of the location sensor, and the step sizes chosen to quantize the parameters. These two types of errors can be traded off against each other by varying the quantization steps. The tradeoff

of one location-dependent parameter is depicted using the receiver operating curve (RoC), illustrated in Figure 2.2. Each different curve in the plot represents an RoC with a different absolute distance in a parameter between x , an authentic user's measurements, and y , an attacker's measurements. A more secure system aims for a low P_{FA} at the expense of a high P_{FR} , whereas a more convenient system aims for a low P_{FR} at the expense of a high P_{FA} .

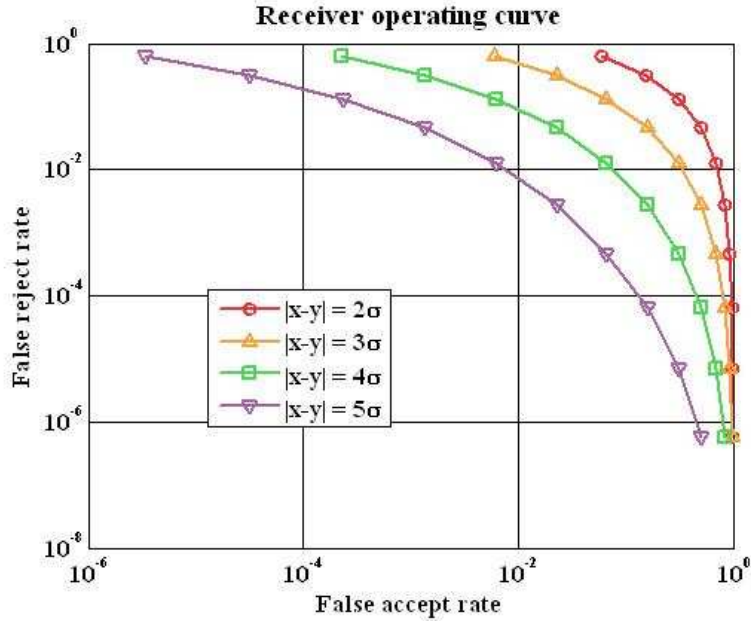


Figure 2.2: Tradeoff between P_{FR} and P_{FA}

2.2.2 Spoofing Attacks

Spoofing is an attempt to circumvent system control by claiming to be an authentic user using false data or signals. A system vulnerability is a design flaw or feature that creates a security weakness, thus presenting an opportunity for attacks. Security functions as a chain, and a single weak link can break the entire system. As a result, it is necessary to analyze the vulnerability in every single link and defeat spoofing attacks one by one to guarantee a desired level of security. An attack model for spoofing is depicted in Figure 2.3. The red arrows with labels represent generic

cryptographic attacks, which are discussed below.

1. Insecure RF signals. The first weakness in the geo-security algorithm is the fact that the broadcast signals are not secure. An attacker or unauthorized user can simulate RF signals to pretend that they are at the location where the legitimate user is. The purpose of geo-security is to provide more security to the security system. As such, it is important that every link of the geo-security system chain is secure. This security includes not only the protocol itself but also the broadcast signals. Message authentication can provide the security of the signal by preventing a user from being fooled into believing that a message comes from a particular source when it is not the case.
2. Tamper with signals. Attackers may use authenticated signals from real transmitters to bypass the signal source verification but modify the received signals and replay them to spoof the authentication device. After the location sensor, the signals are digital. This attack requires the attackers to possess signal processing skills to modify the location-dependent parameters carried on the RF signals. For instance, attackers can re-position RF pulses to modify the time-of-arrival as well as other parameters of the incoming signals.
3. Brute force attack. Attackers generate all the possible combinations of the binary tags and replace the geotag computed from the received location information with the generated geotags. To protect against such an attack, it is desired to have location-dependent parameters with high information entropy that result in long geotags.
4. Tamper with the geotag database. The attacks on the template database include adding a new template, modifying an existing template, removing templates, copying template data for secondary uses, etc.
5. Man-in-the-middle attack. The transmission medium between the template database and the matcher is similarly vulnerable. This attack can result in an alteration of the transmitted template.
6. Override the final decision. Results from the matcher (accept or reject) can be overridden by the attackers. If this is a Denial of Service (DoS) attack, attackers

can change the final decision by producing a sufficiently large number of errors to the system.

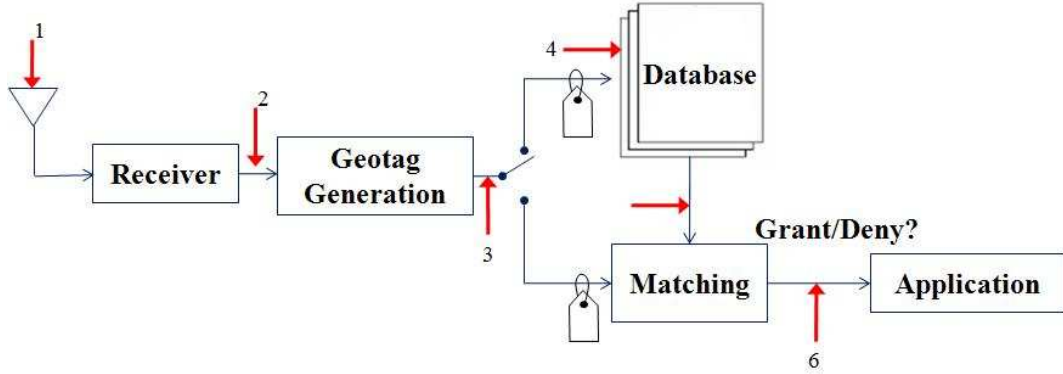


Figure 2.3: Geo-security attack model

2.2.3 Defense against Spoofing

Having understood the threat vectors or the links that are vulnerable to generic attacks, two important requirements must be achieved to protect against spoofing. They are tamper-resistant devices and self-authenticated signals.

2.2.3.1 Tamper-Resistant Device

The spoofing attacks of Type 2 through 6 can be defeated using a tamper-resistant device. A tamper-resistant device, which is commonly used in the security world, offers physical protection to sensitive information residing within the device, thereby providing some assurance that the information cannot be maliciously read or modified. These devices can be used for a wide range of security applications.

There are different ways to achieve the tamper resistance function in practice [1]. A typical tamper-resistant enclosure for protecting an electronic device containing sensitive information (e.g., an electronic cryptographic card) includes an external cover and an internal cover. Screws are applied to secure the internal cover to the device and the external cover by means of standoffs. The device has a dielectric

substrate with a metal pad around each hole to ensure the electrical connection to ground through the washer, the standoff, and the conductive body of the external cover. When one of the screws is partially removed and the internal cover is moved, the circuit is interrupted and the system assumes that tampering is being attempted. The circuit is designed to sound an alarm and/or destroy all the sensitive information contained in the protected electronic device.

2.2.3.2 Signal Authentication Scheme - Timed Efficient Stream Loss-tolerant Authentication

The tamper-resistant device does not protect against the simulated signal attack, Type 1 in Figure 2.3. The safest defense against such a spoofing attack is cryptographic authentication for the signal itself. The principal challenge of a secure broadcast communication is source authentication, which is complicated by untrusted or uncertified users and unreliable communication environments. The concern is that untrusted users may employ items such as signal simulators to spoof the system into generating the correct geotag. Source authentication helps the receivers to verify whether the received data originating from the source have been modified in transit. Furthermore, adding security to a broadcast communication system is difficult because symmetric authentication algorithms are fast and efficient, but are not as secure as asymmetric ones in a broadcast setting; on the other hand, the asymmetric authentication algorithms are secure, but not efficient.

Timed Efficient Stream Loss-tolerant Authentication (TESLA) [33] is considered to be embedded in navigation signals to prevent spoofing. TESLA is a data authentication scheme to ascertain the origin of the incoming signal and allow receivers to verify the integrity of the navigation data messages. TESLA uses symmetric authentication mechanisms by appending an authenticated message at the end of each navigation data message, which is broadcast from a transmitter to a receiver. It also uses delayed key disclosure to achieve the asymmetry that is required for a secure broadcast authentication. The main features of TESLA are: low sender and receiver computation overhead, low communication overhead, and robustness to message loss. Message buffering is required for both the sender and receiver sides, but the

receiver can authenticate the message as soon as enough messages, keys and MACs are buffered. A MAC is computed from data messages and a key by the sender. Once a receiver buffers enough messages, a new MAC is computed from the received data messages and the key, and verified with the received MAC to authenticate the data source. The authentication performance significantly depends on the received signal-to-noise ratio (SNR), the data channel capacity, and the length of the authentication message. Attackers cannot simulate or spoof self-authenticated signals because they do not have the disclosed keys.

To implement TESLA, it is required that a transmitter and a receiver should be loosely synchronized in time. The synchronization does not have to be precise, but the receiver has a rough notion of the sender's local time. Therefore, a secure time channel is required for receivers, either using Internet or Loran time messages to achieve this goal. The following is the outline and sketch of the TESLA approach [33].

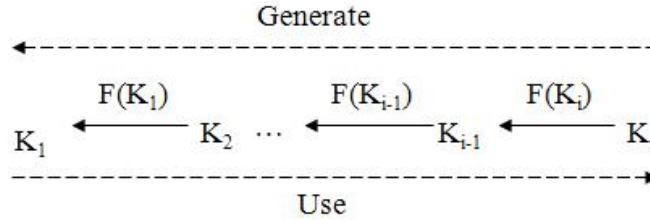


Figure 2.4: One-way key chain generation

1. One-way key chain generation: A TESLA chain of size N is selected. The transmitter generates a one-way chain of N self-authenticating values or keys, denoted K_1, \dots, K_N , and assigns the keys to the N segments (one segment is the time interval necessary for one authentication message) sequentially. A hash function is used to construct the one-way chain, derived from the base key, K_N . The other keys, K_i , are generated from $N - i$ hashes of K_N . Notationally, $K_i = F(K_{i+1}) = F(F \dots F(K_N))$ where there are $N - i$ instances of the hash function, F . Figure 2.4 illustrates the construction of a one-way key chain, and

F indicates the hash function used. When the keys are broadcast, the chain is sent in the reverse order of generation.

2. MAC key generation: The transmitter uses a different hash function, F' , to hash the last one-way chain values and results in the keys, K'_1, \dots, K'_N , used to form MACs.
3. Broadcast stage: The messages, keys and MACs are transmitted in segments. Each segment consists of a message, a MAC and a key associated with the message in the previous segment, illustrated in Figure 2.5.
4. Key verification: Each receiver buffers the segments first. The first step is to verify the received key's values. This is accomplished by hashing the key in the current segment and comparing it with the key in the previous segment.
5. MAC verification: Each receiver checks the correctness of MAC of buffered segments after the keys have been verified. If the MAC is correct, the receiver accepts the segment.

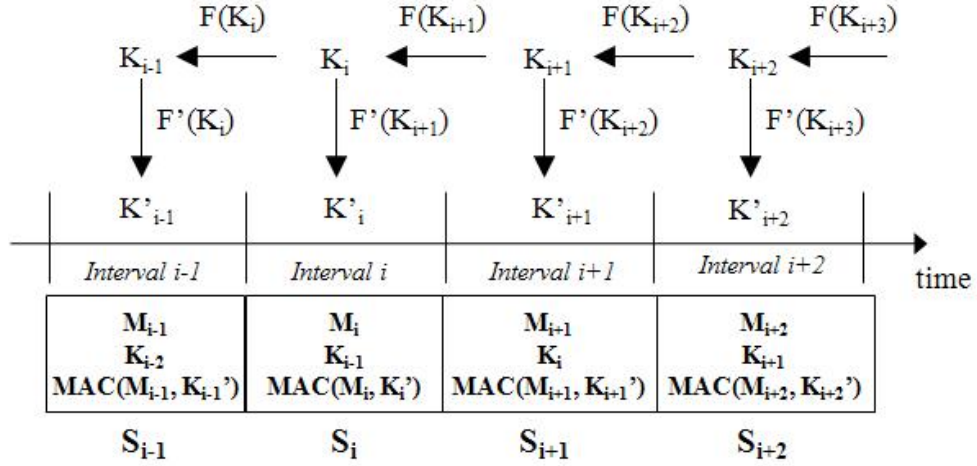


Figure 2.5: TESLA setup

Once the signal source is authenticated, that is, the incoming signals are confirmed to be from real transmitters rather than a spoofer or a signal simulator, the receiver proceeds to extract location-dependent parameters and compute a geotag. The signal

authentication can also detect modified or changed data messages modulated on the signals, thus ensuring the data integrity.

2.2.4 Location-Based Attacks

2.2.4.1 Location Brute Force Attack

This section provides thorough analyses on non-generic cryptographic attacks in geo-security systems to complete the threat model. With a tamper-resistant device and self-authenticated signal, attackers cannot employ the spoofing attacks seen in Figure 2.3. To bypass the location validation or to achieve an authentic geotag, attackers have to be physically close to a legitimate user's location, illustrated in Figure 2.6. Since the user's location might not be secret and there is no physical boundary to distinguish an authorized user from an attacker, an attacker might be able to break the system by trying the possible locations to which he has access. This is called a *location brute force attack* or *simple attack*. In conventional cryptanalysis, a brute force attack is a method to defeat an algorithm or protocol by systematically trying a large number of possibilities [47]. Thus, an appropriate key length is chosen to make a brute force attack computationally infeasible. In an instance of location brute force attack in geo-security, an attacker can try all of the possible locations to which he has access and accidentally achieve an authentic user's geotag by mapping real signal characteristics associated with these locations collected using the tamper-resistant device.

A location brute force attack relies on probabilistic mapping from the attacker's location to the user's. The probability of a false accept is used to measure the difficulty of the location brute force attack. The P_{FA} error bound is estimated using a Bayesian approach and considering the *a posteriori* error distribution. Let x be the location-dependent vector of an authentic user, and y be the attacker's. Considering one location-dependent parameter, the probability of a false accept is represented as:

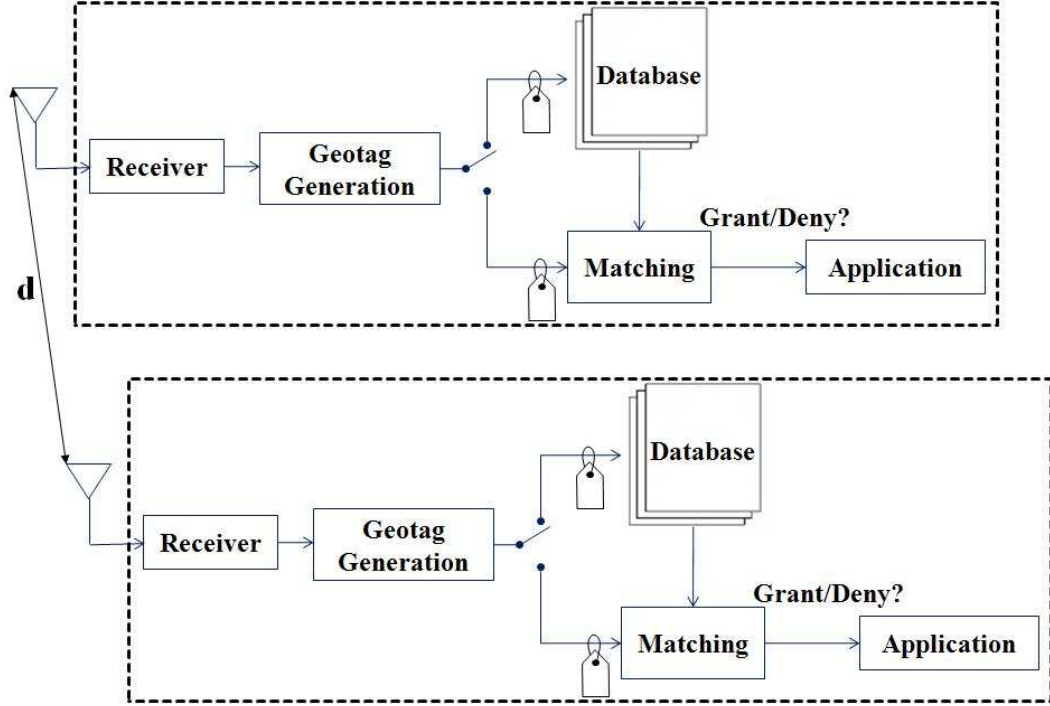


Figure 2.6: Location brute force attack

$$\begin{aligned}
 P_{FA} &= Q\left(\frac{|x-y| - \frac{\Delta}{2}}{\sigma}\right) - Q\left(\frac{|x-y| + \frac{\Delta}{2}}{\sigma}\right), \\
 Q(\alpha) &= \frac{1}{\sqrt{2\pi}} \int_{\alpha}^{\infty} e^{-\frac{x^2}{2}} dx,
 \end{aligned} \tag{2.14}$$

where $|x - y|$ is the absolute distance of a parameter between user and attacker, and the Q -function is a convenient way to express tail probabilities for Gaussian random variables. $|x - y|$ can be proportional to the physical distance between the user and the attacker, the standard deviation of the parameter, but inversely proportional to the decorrelation distance of the parameter, d_0 . A decorrelation distance is defined as the physical distance at which the attacker's P_{FA} is reasonably small and measures the difficulty of the location-based attacks. As a result, P_{FA} can be written as a function of normalized distance, $\frac{d}{d_0}$.

$$P_{FA} = Q\left(\frac{d}{d_0} - \frac{\Delta}{2\sigma}\right) - Q\left(\frac{d}{d_0} + \frac{\Delta}{2\sigma}\right). \quad (2.15)$$

The false accept rate of one parameter with a typical $\Delta = 4\sigma$ is plotted in red in Figure 2.7. With more parameters that are assumed to be uncorrelated with one another to compute a geotag, the P_{FA} is reduced dramatically as the spatial decorrelation of the derived geotag improves.

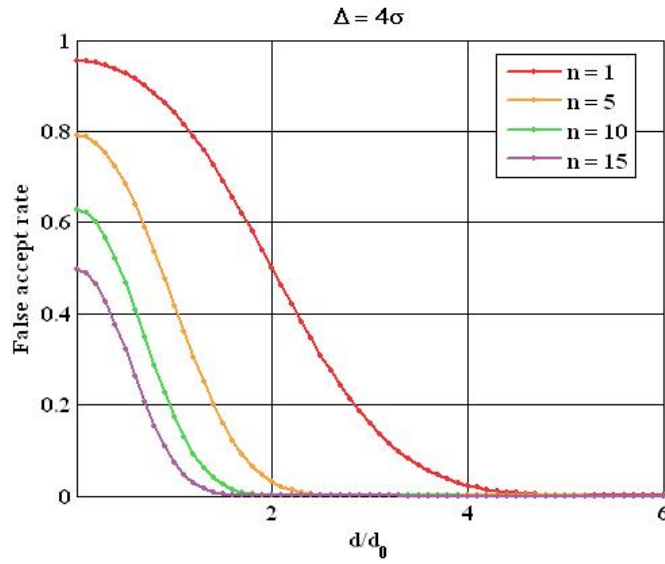


Figure 2.7: Spatial decorrelation of uncorrelated parameters

In practice, the location-dependent parameters are correlated to a certain degree. Now considering the correlation between different location-dependent parameters, the relation between x and y is represented by $y = Ax + b + v$, where A is an $n \times n$ matrix, b is the difference or bias vector of location-dependent parameters from two different locations, and v is the random variable representing the estimation error with zero mean and standard deviation σ . All the vectors are n -dimensional. The next step is to investigate the probability density of errors affecting the attackers' estimations on x . Let $z = b + v$. The error probability density error of each individual location-dependent parameter z_i can be written as:

$$p(z_i) = f_{b_i, \sigma_i}(z_i). \quad (2.16)$$

The distribution is Gaussian with mean b_i and standard deviation σ_i . The joint density of n location-dependent parameter errors or a mixture of multivariate Gaussian is

$$p(z_1, z_2, \dots, z_n) = B|W|^{1/2} \exp \left(-\frac{1}{2} (z - b)^T W (z - b) \right), \quad (2.17)$$

where C is the covariance matrix, $W = C^{-1}$ and $B = (2\pi)^{-n/2}$. The covariance can be computed from either actual measurements or modeling. The probability density of the authentic user's measurements, x , is evaluated given the attackers' measurements, y . Consider the *a posteriori* error distribution using the Bayesian approach, that is,

$$p(x|y) = \frac{p(x, y)}{p(y)} = \frac{p(y|x)p(x)}{p(y)}. \quad (2.18)$$

Here $p(x)$ and $p(y)$ are the *a priori* distributions, which are considered to be a uniform distribution over the whole space. Let $\hat{y} = y - b$. Having z as a mixture of Gaussian distributions, $p(x|y)$ can be expressed as:

$$p(y|x) = B|W|^{1/2} \exp \left(-\frac{1}{2} (\hat{y} - Ax)^T W (\hat{y} - Ax) \right). \quad (2.19)$$

After some algebra, $p(y|x)$ is found to be

$$p(y|x) = B|W|^{1/2} \exp \left(-\frac{1}{2} \hat{y}^T (W - WA\Sigma^{-1}A^TW) \hat{y} \right) \times \exp \left(-\frac{1}{2} (x - \hat{x})^T \Sigma (x - \hat{x}) \right) \quad (2.20)$$

where $\Sigma = A^TW A$, and $\hat{x} = \Sigma^{-1}A^TW \hat{y}$. Note that one of the terms in Equation (2.20) is a Chi-square distribution, such that $\chi^2 = \hat{y}^T (W - WA\Sigma^{-1}A^TW) \hat{y}$. Compute $p(y)$ by integrating $p(y|x)$ over all of the possible locations,

$$\begin{aligned} p(y) &= \int_x p(y|x)p(x)dx \\ &= B|W|^{1/2} e^{-\frac{1}{2}\chi^2} \times \int_x \exp \left(-\frac{1}{2} (x - \hat{x})^T \Sigma (x - \hat{x}) \right) p(x)dx. \end{aligned} \quad (2.21)$$

The integral term can be solved analytically,

$$\int_x \exp\left(-\frac{1}{2}(x - \hat{x})^T \Sigma (x - \hat{x})\right) dx = \sqrt{\frac{(2\pi)^n}{|\Sigma|}}. \quad (2.22)$$

Now by plugging Equations (2.20), (2.21), and (2.22) into (2.18), the *a posteriori* error distribution is given by

$$p(x|y) = p_{\hat{x}, \Sigma^{-1}}(x), \quad (2.23)$$

where \hat{x} is the mean, and Σ^{-1} is the covariance matrix of the multivariate Gaussian distribution. Finally, the generalized probability of a false accept in Figure 2.1 can be written as:

$$P_{FA} = \int_{\hat{x}-\Delta/2}^{\hat{x}+\Delta/2} p(x|y)p(y)dx. \quad (2.24)$$

Based on Equation (2.24), the probability of a false accept significantly depends on the parameter spatial decorrelation, the Euclidean distance of the location-dependent parameters between the user and the attacker, the quantization steps chosen, and the Gaussian statistics of the measurements. The higher the spatial decorrelation, the lower the false accept rate. The parameter variation results from a combination of many factors, such as the quality of antenna and receiver, local noise floor, propagation path, and atmospheric noise.

2.2.4.2 Selective Delay Attack

In many RF systems, an antenna is not tamper-resistant. To guarantee signal reception, an antenna is usually placed in an open sky area to avoid fading, signal blockages, or reflections. Such a setup allows attackers to insert a delay loop between the antenna and the receiver to tamper with the received signals, depicted in Figure 2.8. The delay loop will fool the receiver into thinking that it is somewhere else, particularly the authentic user's location. The function of the delay loop is to allow attackers to manually modify location-dependent parameters by processing the analog signal, such as lowering or increasing the received signal strength or SNR, advancing or delaying signal phase and time-of-arrival (TOA). Separate delay loops are required

for each channel or signals from each transmitter to perform the analog signal processing. As a result, more location-dependent parameters from multiple transmitters increase the complexity and lower the successful rate of the selective delay attack.

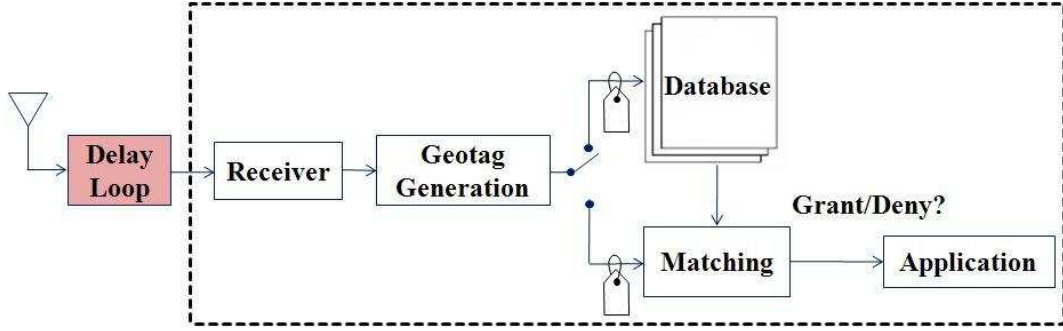


Figure 2.8: Selective delay attack

Two possible types of selective delay attacks are illustrated in Figure 2.9. The attacks require additional spoofing hardware: a spoofer and a delay-loop module. To perform these attacks, attackers require some degree of technical skill, such as signal processing.

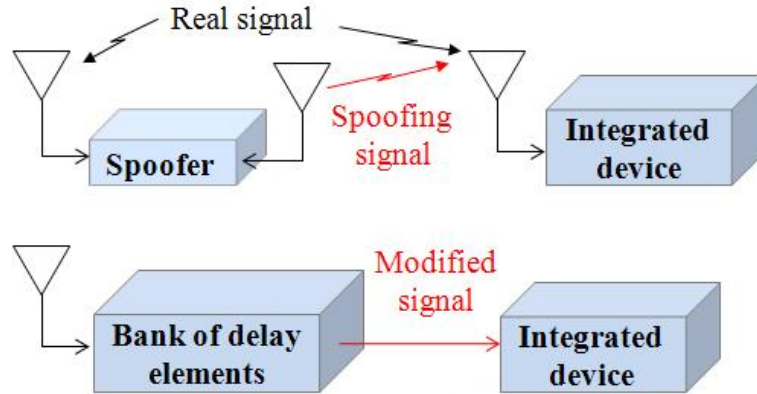


Figure 2.9: Illustration of spoofing attacks in action

As one might imagine, there are no commercially available delay-loop modules. This, of course, decreases the present likelihood of the selective delay attack. Another challenge of the attack is that the attacker has to gain accurate knowledge of the target or the authentic user's position to be able to match the target geotag. In

addition, the tamper-resistant device increases the uncertainty or randomness of such guessing since attackers cannot open the black box to learn the exact values of the location-dependent parameters.

The further an attacker is from the target location, the more numerous the trials or the time it takes for an attacker to break into the system. An upper bound for the attacker's total number of guesses or trials can be derived; it is given by

$$N_{trial} = \sum_{i=1}^n w_i \binom{n}{i} \prod_{j=1}^i m_j, \quad (2.25)$$

where w_i is a weighting factor that depends on the attacker's knowledge of the target user's location information, n is the total number of parameters to compute a geotag, and m_j is the number of possibilities to modify a location-dependent parameter. For example, if the quantization level difference between the user and an attacker is 1, there are two possibilities $(1, -1)$ for the attacker to guess. Note that N_{trial} is the maximum if w is uniformly distributed which implies that the attacker has little or no knowledge on the target user's location-dependent parameters. Figure 2.10 illustrates how N_{trial} changes with m and n , assuming that $w = [1, 1, \dots, 1]^T$. For instance, if $m = 3$, and $n = 11$, the upper bound of the number of trials is 4,194,303.

The lower bound can be seen as the case in which an attacker can predict correctly which location-dependent parameters differ. In other words, the attacker does not need to go through the $\binom{n}{i}$ combination search, and w is a unit vector. As a result, the lower bound is simplified as $\prod_{j=1}^i m_j$. It is not easy for attackers to guess the difference in the parameter values between the user and the attacker due to the uncertainty originating from the signal attenuation, reflection, and refraction.

2.3 Continuity

Continuity becomes a problem when a user is warned that current conditions do not allow him to proceed with an application due to an interruption of the authentication service. The service interruption may happen unexpectedly due to a failure of

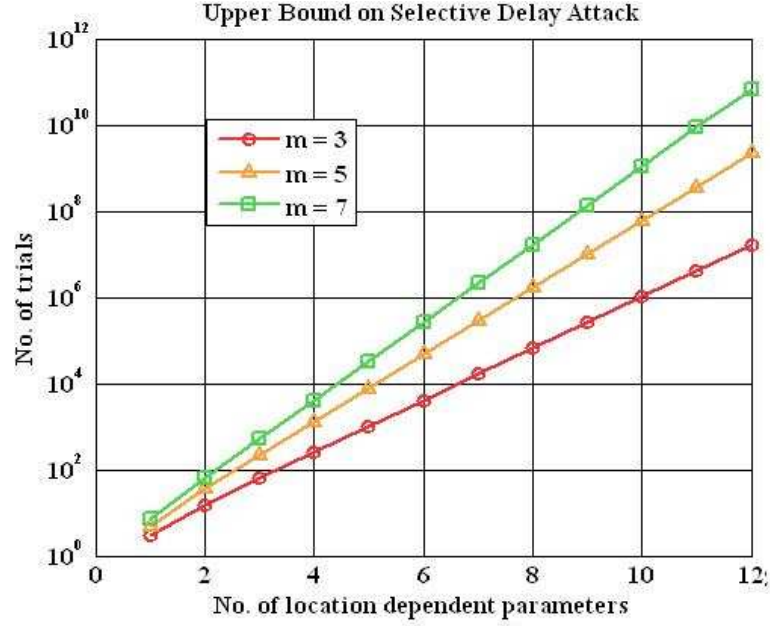


Figure 2.10: Selective delay attack: upper bound on the number of trials

the radio transmitter that is broadcasting the signals used for authentication; or by receiver errors that push the measurement outside of the ranges that correspond to the geotag. Continuity risk is defined as the probability that a verification procedure fails due to a loss of service for either of these reasons. The false reject rate, P_{FR} , discussed in Section 2.2.1 is utilized to measure continuity risks.

A loss of continuity on the radio transmitter side has an inconvenience cost resulting from the fact that one or more transmitters are offline due to maintenance or other practical issues. A simple continuity assessment is possible from the perspective of signal seasonal monitoring. A monitor station periodically takes a snapshot of signal information from the monitored transmitters and decides whether the extracted signal characteristics from the transmitters are good enough to compute a geotag. If the system fails to authenticate a legitimate user, then a *false detection* has occurred. On the other hand, if the system authenticates an attacker, then a *missed detection* has occurred. A false detection will lead to a failure of an authentic user to validate his location.

A loss of continuity on the receiver side can result from a receiver shutdown or an

improper design in the geotag generation. For instance, if a chosen quantization step is small, failing to overbound the variation of location-dependent parameters increases users' false reject rates. Thus, temporal variations of location-dependent parameters reduce repeatable accuracy, in other words, degrade the reproducibility of a geotag. Assuming that the parameters have a Gaussian distribution, the analytical probability of a false reject can be calculated as

$$P_{FR} = 2Q\left(\frac{\Delta}{2\sigma}\right). \quad (2.26)$$

Thus, a quantization step can be estimated by inverting Equation (2.26),

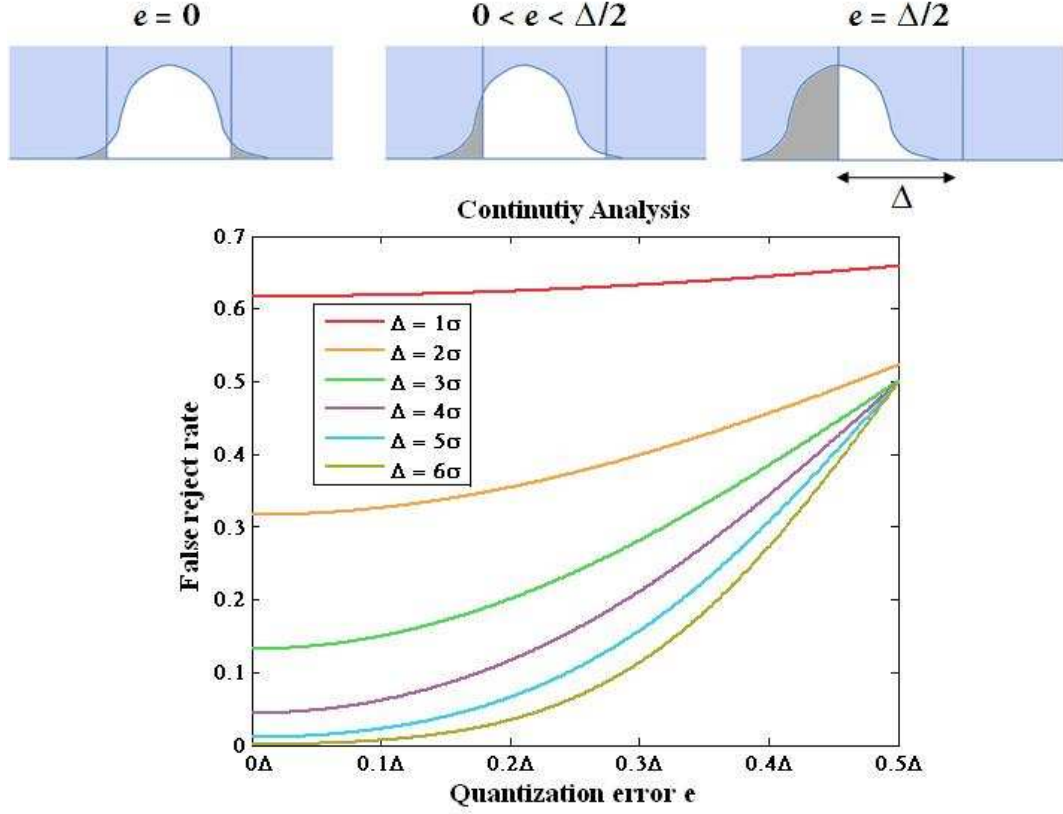
$$\begin{aligned} &\text{minimize} \quad \Delta, \\ &\text{subject to} \quad P_{FR} = 2Q\left(\frac{\Delta}{2\sigma}\right). \end{aligned} \quad (2.27)$$

The quantization error, e , which is the difference between a quantized value and a true measurement, also increases the continuity risks. The P_{FR} with non-zero quantization error can be written as:

$$P_{FR} = Q\left(\frac{\frac{\Delta}{2} + e}{\sigma}\right) + Q\left(\frac{\frac{\Delta}{2} - e}{\sigma}\right). \quad (2.28)$$

Figure 2.11 gives the relation between the probability of a false reject, P_{FR} , quantization error, e , and quantization step, Δ . The best scenario is one in which there is no quantization error, that is, the true measurement is in the middle of a quantization grid. The worst case, on the other hand, is the case in which $e = \frac{\Delta}{2}$, that is, a user will have 50% likelihood of failing to validate his location.

To improve continuity, users should select larger quantization steps to achieve low false reject rates. Based on Equation (2.27), a quantization step depends on the parameter standard deviation, σ , and desired P_{FR} , which is a design parameter. The standard deviation can be estimated using either monitored data or modeling. The variation of location-dependent parameters is inversely proportional to the received signal-to-noise ratio. Of course, increasing the quantization steps leads to an increase in the probability of successful attack, that is, the tradeoff curve in Figure 2.2 shifts

Figure 2.11: False reject rate as a function of quantization error e

from upper left to lower right.

2.4 Temporal and Spatial Entropy

This section develops a mathematical framework to measure location-dependent information entropy, quantify geotag repeatability and spatial decorrelation, and compare the strengths of different parameters. In this dissertation, spatial decorrelation measures the changing rate of a geotag derived from location-dependent parameters as a function of physical location.

Information measures play a significant theoretical role in connection with cryptographic systems. For instance, entropy-based arguments can provide a way to quantify repeatable accuracy and spatial decorrelation as well as an upper bound on

the geotag length in geo-security systems.

2.4.1 Temporal Entropy to Measure Continuity

Consistency is a key requirement for location-based security. Temporal entropy or instability is a metric to measure the time stability of location-dependent parameters. Parameter variation reflects the instability or degree of scatter of a particular parameter at a given location. For geo-security, parameter stability with low temporal entropy is a fundamental requirement. For a particular quantization, the larger the temporal variation, the higher the likelihood that an authorized user will fail to generate the correct geotag. Many factors can result in temporal entropy. Some factors are related to the receiver or algorithms employed and proper design can eliminate these variations. Others are related to propagation and changes in the environment. They are not so easy to eliminate and must be well understood.

The temporal entropy, H_T , can be computed using Equation (2.29) for a given probability distribution of any quantized received location-dependent parameter. Equation (2.29) is also called *Shannon entropy* or *information entropy* [14]; additional reviews of information theory are discussed in Appendix A. The temporal entropy is inversely proportional to the parameter quantization step. To ensure that the user is able to authenticate successfully, the quantization step should be adequate, but cannot be excessively large, since it will increase an attacker's false accept rate and reduce the total information entropy.

$$H_T(X) = - \sum_{x \in X} p(x) \log p(x). \quad (2.29)$$

2.4.2 Spatial Entropy to Measure Integrity

The uniqueness of a parameter for geo-security is quantified using spatial entropy, H_S . For parameters with low spatial entropy, users in different locations will measure similar or identical values. Higher spatial entropy helps provide more uniqueness to the geotag for users at different locations. Therefore, larger spatial entropy results in a stronger geotag and a higher security level for the system. Spatial entropy is also

an indicator for the quantitative information capacity of each location-dependent parameter.

As mentioned in Subsection 2.2.1, the problem is modeled as hypothesis testing. Let the two hypotheses, H_0 and H_1 , have the probability distributions P_{H_0} and P_{H_1} . The two possible errors that can be made in a decision are $\alpha = P_{FR}$ – accepting hypothesis H_1 when H_0 is true and $\beta = P_{FA}$ – accepting H_0 when H_1 is true. The relative entropy [14], represented as $D(P_{H_0}||P_{H_1})$, is used to estimate the spatial entropy that is the information used to distinguish the two probability distributions by hypothesis testing. Let $d(\alpha, \beta)$ be

$$d(\alpha, \beta) = \alpha \log \frac{\alpha}{1 - \beta} - (1 - \alpha) \log \frac{1 - \alpha}{\beta} \quad (2.30)$$

$$d(\alpha, \beta) \leq D(P_{H_0}||P_{H_1}). \quad (2.31)$$

If a proper step size is chosen, small α can be achieved, that is, $\alpha \ll 0$, and H_S is

$$P_{FA} \geq 2^{-D(P_{H_0}||P_{H_1})} \quad (2.32)$$

$$H_S = D(P_{H_0}||P_{H_1}) \geq -\log P_{FA}. \quad (2.33)$$

Therefore, spatial entropy can be computed using an attacker's false accept rate from Equation (2.33), which yields a theoretical lower bound of the uniqueness measure. Furthermore, temporal and spatial entropies are connected. High temporal variability forces the designs to increase Δ . As Δ approaches desired decorrelation distance, the spatial entropy decreases.

2.4.3 Information Entropy to Bound Geotag Length

Information entropy or information density is an indicator for the quantitative information capacity of each location-dependent parameter. High information entropy indicates a large potential value space of a geotag. The potential information density depends primarily on the transmitter coverage as well as the quantization step of the

parameter. Technically speaking, the information bits in a computed geotag can be estimated based on the potential information capacity of the parameters. In practice, the effective amount of entropy in a geotag can be reduced if a selective delay attack is performed.

Assume that the quantized location-dependent parameters are uniformly and independently distributed. Applying Shannon entropy, the total information entropy or geotag size in a geo-security system can be interpreted as

$$H = \sum_{i=1}^n (\log N_i - H_{T_i}), \quad (2.34)$$

where n is the total number of parameters, and N_i is the possible occurrences of each individual parameter after quantizing the parameter with a particular step size. Temporal entropy, H_T , increases the probability of a false reject, degrades the reliability of the system, and reduces the information entropy of the total system.

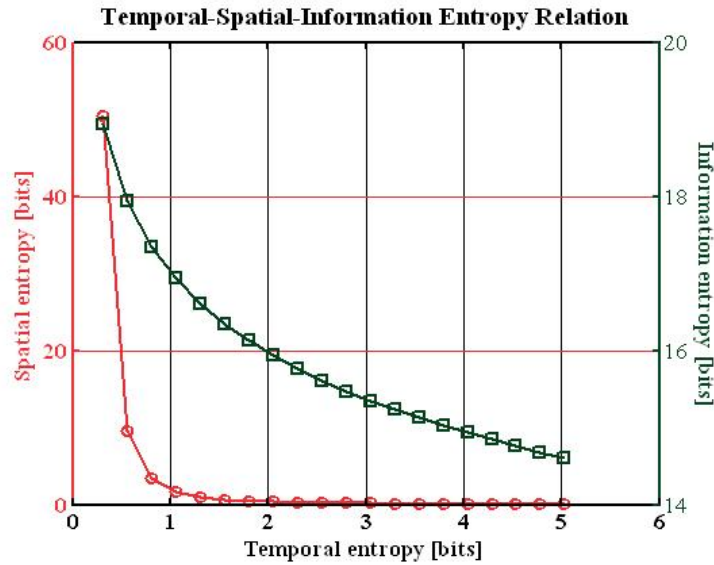


Figure 2.12: A reduction in spatial entropy and total information with an increase in temporal entropy. The temporal entropy is derived by varying the quantization steps.

The relationship between temporal entropy, spatial entropy and information entropy of one location-dependent parameter is illustrated in Figure 2.12. The change

of temporal entropy has a significant impact on the spatial entropy and the information bits in a geotag. Large temporal entropy indicates high temporal instability and unreproducible geotags; high spatial entropy and information entropy, which are the desired properties, indicate a low false accept rate and more secure system.

2.5 Conclusions

This chapter designed and modeled a geo-security system that uses location-dependent signal characteristics rather than physical locations mapped into a geotag to provide an additional layer of security. Compared with latitudes, longitudes and heights, location-dependent signal characteristics are more unpredictable and provide more information entropy, which results in a longer geotag, which are more robust to the brute-force attack. In addition, a standard process and theoretical framework to evaluate the system performance were developed. Two performance standards –continuity and integrity– measure the reproducibility and security strength of a derived geotag, respectively.

A false reject rate (FRR), P_{FR} , and a false accept rate (FAR), P_{FA} , quantify the continuity and location-based brute force attack. A threat model, which defines the possible spoofing attacks due to system vulnerabilities as well as implementation weaknesses and the location-based attacks, was built to analyze and quantify the geo-security integrity. A tamper-resistant device and self-authenticated signals are required to protect against spoofing attacks. An upper bound on the number of trials for an attacker to break into the system is estimated.

The trade space between continuity and integrity is quantified using FAR and FRR by varying the location-dependent parameter quantization steps. A more secure system aims for low FARs at the expense of high FRRs, whereas a more convenient system aims for low FRRs at the expense of high FARs. The desired quantization step significantly depends on the final application.

Chapter 3

System Design for Loran

The most important feature required for a signal to demonstrate geo-security is the ability to generate a strong geotag. As mentioned, the strength of a geotag is determined by the quantity and quality of location-dependent signal parameters. The *quantity* refers to the number of different location-dependent parameters that can be extracted. The *quality* corresponds to the amount of unique location-dependent information provided by each parameter. The information content is connected to the repeatability and spatial decorrelation of the parameters. For example, greater spatial decorrelation results in more unique information. By receiving many parameters with each providing unique information content, a strong geotag can be generated.

At the same time, it is desirable to select parameters that are relatively insensitive to temporal changes which weaken the robustness of geo-security systems. Temporal variations essentially reduce the uniqueness of the location-dependent information. As a result, repeatable accuracy is a desirable quality which allows a user to receive his location-dependent parameters or the derived geotag at one time—and still have those parameters valid at a later time. In other words, the signal characteristics should be consistent enough so that when the user is ready to authenticate, measurements at the same location will yield the same previously-generated geotag.

Furthermore, there are several highly desirable features. The signal should have anti-spoofing capabilities. If the signal is vulnerable to spoofing, it may be possible for an attacker to bypass the location check and authenticate correctly. In addition,

the signal should be available indoors because many of the anticipated applications of geo-security will likely occur indoors. Possible applications are digital manner policy, the management and distribution of secure digital data, and laptop security. Often, it is essential that data and electronic devices are only accessible within certain buildings.

The possible signals including RF and non-RF are depicted in Figure 3.1. Examples of non-RF signals are infrared and ultrasound, which provide good repeatable accuracy but small signal coverage. Typical infrared and ultrasound coverage is less than the size of a room. The interference sources of these signals include the room temperature and indoor lights. Although the most commonly used navigation system today is satellite-based, also referred to as *GNSS*, its extremely weak signals originating from the long propagation distance from satellites to the Earth results in signals that are easy to jam and spoof. Such a vulnerability leaves a weak link for attackers to break geo-security systems. In contrast, Loran, TV, and GSM systems use high-powered terrestrial signals. Such high radiation powers make these systems robust to jamming. Although TV and GSM have gained in popularity for indoor positioning over the years, it is difficult to achieve accuracies comparable to outdoor GPS. Even so, these could be considered for future research. The remainder of the signals, such as Bluetooth, Wi-Fi, UWB, and RFID, are short-range RF-based and adaptable to indoor applications as well. Finally, Loran has been chosen as a case study for an indepth analysis of geo-security in this chapter.

The outline of this chapter is illustrated in Figure 3.2. To navigate this work more easily, the heart of the chapter is presented in red while the review of Loran is provided in blue. The integrity, continuity, and information entropy analyses with experimental results are discussed in Sections 3.2, 3.3 and 3.4, respectively.

3.1 Loran Background

Loran, developed in World War II, is a terrestrial, low-frequency, pulsed-navigation system that operates in much of the northern hemisphere (see Figure 3.3) [3]. It has numerous properties that are useful to geo-security applications. This section

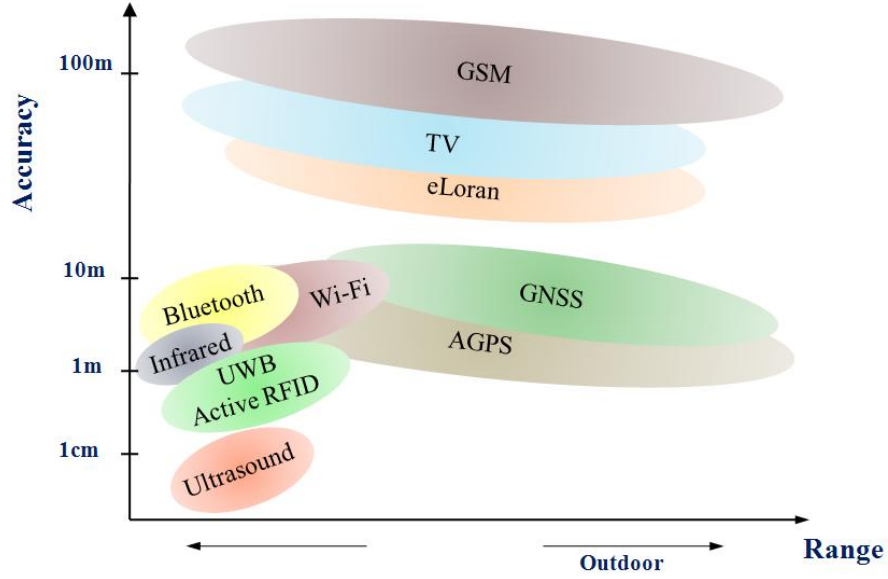


Figure 3.1: Signal opportunities for geo-security

describes the basic Loran system architecture, which is the prerequisite for understanding the geo-security implementation using Loran.

3.1.1 System Configuration

As shown in Figure 3.4, the current Loran system uses 24 stations over the Conterminous United States (CONUS) and 6 additional stations in Canada [2]. The transmitting power of these towers ranges from 350 kW to 1.6 MW. The stations are grouped into 11 chains across the US; each consists of a master station and several secondary stations. Figure 3.4 also highlights the four stations in the Loran West Coast chain: Fallon, NV; George, WA; Middletown, CA; and Searchlight, NV. Most of the testing and experiments in this dissertation used data collected from the West Coast chain.

The Loran stations are synchronized using cesium clocks; timing commands are used to synchronize the transmitters in a chain to broadcast a sequence of pulses at precisely timed intervals. Figure 3.5 illustrates the timing of both the master and secondary stations. The sequence begins with a master station M, which consists

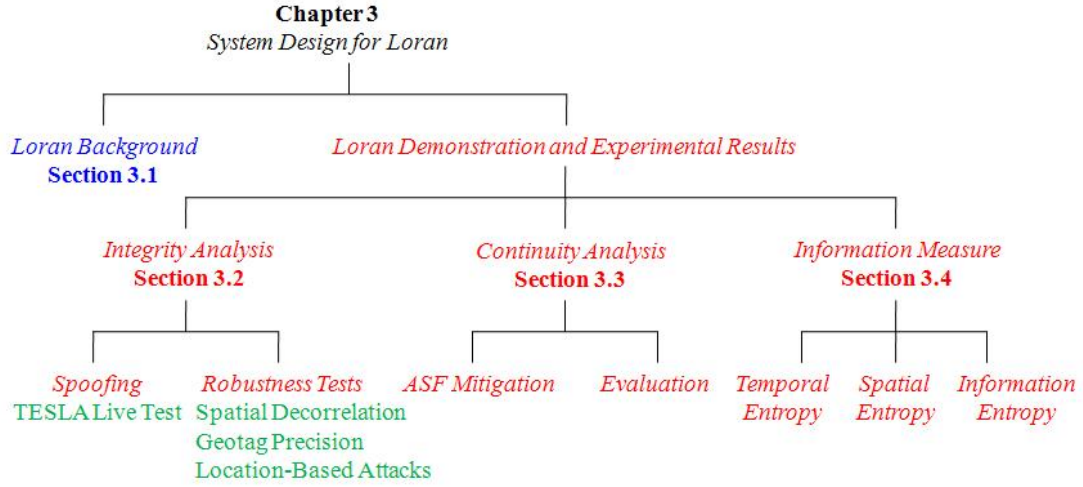


Figure 3.2: Chapter 3 organization

of nine pulses. The first eight pulses are spaced one millisecond apart. The last pulse identifies the master and is spaced two milliseconds from the eighth pulse. The secondary stations, which are referred to by the letters W, X, Y, and Z, have eight pulses. The pulses of each chain are broadcast repetitively at a constant time interval, called the *Group Repetition Interval* (GRI) [3]. The repetition interval of each GRI is designed to be different for the cross-rate interference rejection. The West Coast chain is GRI 9940; thus, its pulse groups broadcast every 0.0994 seconds. Typical GRIs range between 5930 and 9990.

In order to differentiate master pulses from the secondaries, pulse phase coding is applied to Loran pulses. The phase coding repeats every two GRIs, A and B [3]. The interval of two GRI is known as the *Phase Code Interval* (PCI). Loran pulses have an initial phase shift of 0 or 180 degrees, which is equivalent to a sign of + or -. The sequence of the phase code for transmitted pulses is displayed in Table 3.1.

Group	Master	Secondary
A	+ + - - + - + - +	+ + + + + - - +
B	+ - - + + + + + -	+ - + - + + - -

Table 3.1: Loran station phase codes

An ideal Loran pulse with a center frequency at 100 kHz is illustrated in Figure 3.6.

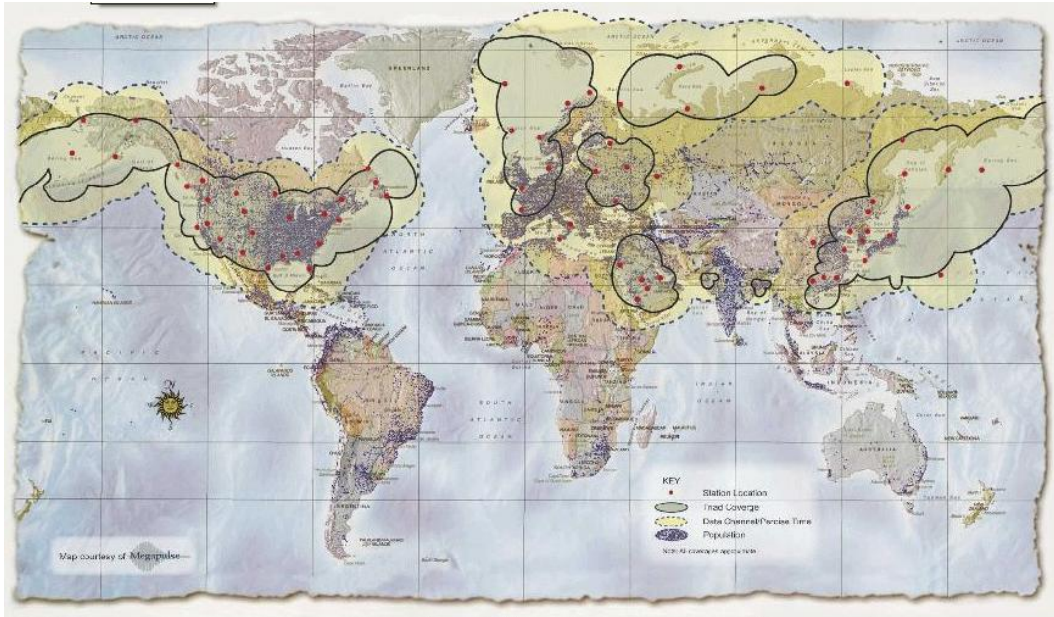


Figure 3.3: Loran world coverage map



Figure 3.4: Loran stations over U.S. and Canada

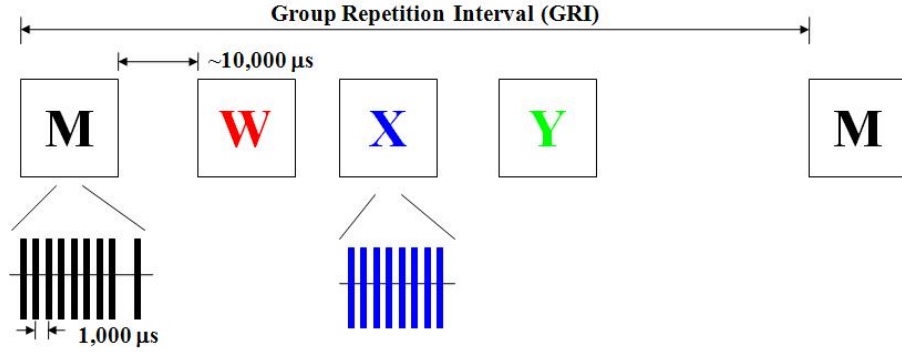


Figure 3.5: Loran GRI timing diagram

It is designed such that the leading edge rises rapidly so that a receiver can capture the tracking point quickly to reject sky-wave interference. On the other hand, the trailing edge decays slowly to shape the signal spectrum [3]. The formula for an ideal pulse is

$$s(t) = (t - \tau)^2 e^{\frac{-2(t-\tau)}{65}} \sin\left(\frac{2\pi t}{10}\right), \quad (3.1)$$

where t is time in microseconds; τ is the time difference between the envelope and carrier referred to as envelope-to-cycle difference (ECD); $(t - \tau)^2 e^{\frac{-2(t-\tau)}{65}}$ represents the envelope of the signal, and $\sin(\frac{2\pi t}{10})$ defines the carrier. Typical values for ECD are in the range of $-5 \leq \tau \leq 5\mu s$. The cycle period of a pulse is $10\mu s$; the pulse peak occurs at the $65\mu s$ point.

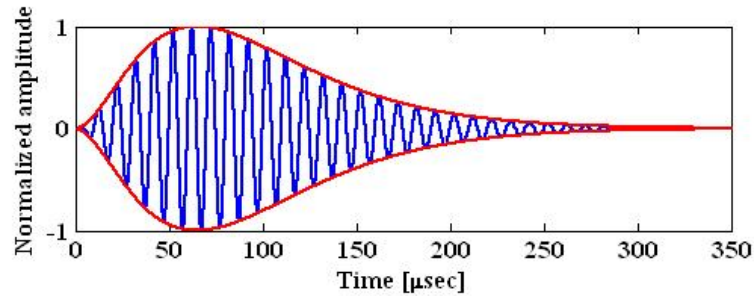


Figure 3.6: Ideal Loran pulse with positive phase code

Rather than using the peak point, a Loran receiver utilizes *third-zero-crossing* as

the *standard-zero-crossing* to detect the signal and track pulses because third-zero-crossing affords reasonably high signal strength and minimizes skywave interference. The identification of a third-zero-crossing can be critical. An erroneous selected cycle will result in an error in one or more time differences of an integer multiple of $10\mu s$, which can cause a range error of 3.0 km. When a transmitter broadcasts, it generates two types of signals: groundwave and skywave. The groundwave propagates over the Earth's surface and becomes more attenuated, whereas the skywave is due to the reflection of signals from the ionosphere down to the Earth. The skywave signal is stronger in amplitude although it travels over a longer distance due to its lower path loss compared with that of the groundwave. The received skywave signal with delay, illustrated in Figure 3.7, may interfere with the groundwave signal and mislead a receiver into tracking an erroneous cycle.

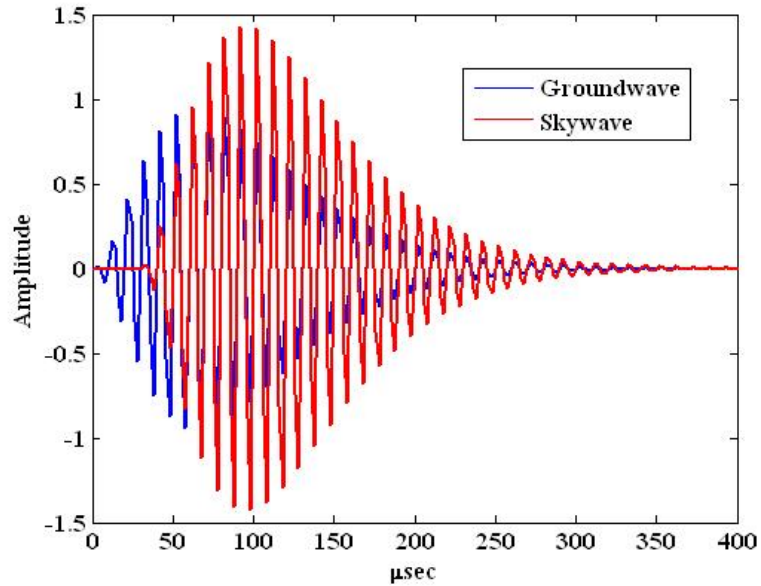


Figure 3.7: Skywave interference

Traditional Loran uses the hyperbolic positioning technique to estimate one's location. The basic idea is demonstrated in Figure 3.8 for 2-D positioning. The red dot indicates a user's location. The dotted lines represent the hyperbolic line-of-position (LOP), determined from the received time difference of each station [2]. A minimum

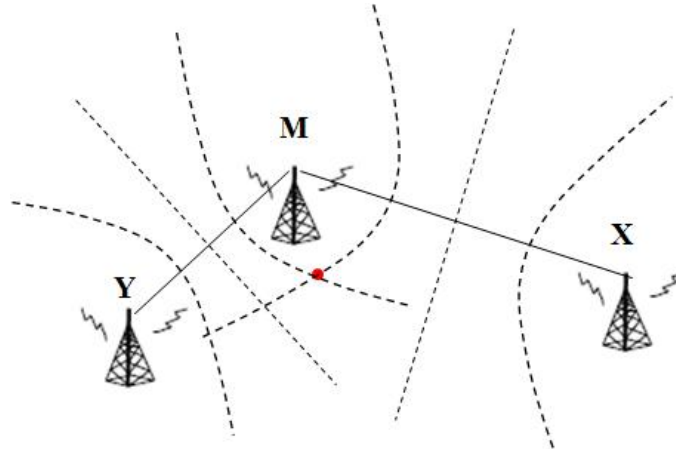


Figure 3.8: Hyperbolic positioning

of three stations are required in order to form two equations to determine the user's latitude and longitude, which are the unknowns here. More stations can be used to resolve ambiguities, thus providing better geometry and accuracy to estimate the position.

3.1.2 Properties Beneficial to Geo-Security Implementation

This section describes the Loran features that can benefit the design and implementation of a geo-security system. First, Loran uses static transmitters; as a result, there are many parameters that are stable in time and location-dependent. Each parameter offers a different amount of information or potential information density. The parameters with higher density result in higher security levels. Such a property is important, since the security strength of the geotag is derived from the information used to generate it. A combination of various parameters as well as the increased accuracy of these parameters, improves the security strength. Signals from static transmitters give many location-dependent characteristics while non-stationary transmitters produce parameters that are not only location-dependent, but also time-dependent. A large temporal component will weaken the security strength.

This dissertation investigates the following Loran signal characteristics:

- time-of-arrival or time difference (TOA/TD);

- envelope-to-cycle difference (ECD);
- absolute or relative signal-to-noise ratio ($\text{SNR}/\Delta\text{SNR}$); and,
- signal strength measured at the peak or third-zero-crossing.

TOA measures the propagation distance from a transmitter to a receiver, whereas the TD of a secondary station is defined as $TD_{s_i} = TOA_{s_i} - TOA_M$. The ECD, resulting from the different propagation speeds of the signal envelope and the carrier, is also location-dependent.

In addition, Loran has good repeatable position accuracy, which benefits the implementation and guarantees the reproducibility of geotags. Furthermore, it is being modernized to a next generation system known as *enhanced Loran* (eLoran), which will have additional capabilities that will further promote its use for geo-security. The broadcast data messages also improve navigation performance.

3.1.3 Loran Data Channel (LDC)

High position resolution and anti-spoofing capacity are the desired signal characteristics for implementing geo-security. Such features can be achieved by enhanced Loran. Types of messages carried in LDC include time, differential phase correction, seasonal bias correction and station information. The LDC not only improves the position resolution for navigation but also opens an opportunity to implement signal authentication for anti-spoofing. The current U.S. proposal uses ninth-pulse modulation. The modulation technique is chosen to minimize the impacts on the current operational Loran signal. An additional pulse is inserted after the eighth pulse of the pulse group of secondary stations [31]. Thirty two state Pulse Position Modulation (PPM 32), illustrated in Figure 3.9, resulting in a data rate of 5 bits/pulse is used to control the time delay of the ninth pulse from 1000 microseconds subsequent to the eighth navigation pulse. The delay of the 32 symbols is given in the formula from zero-symbol offset,

$$d_i = 1.25 \text{mod}(i, 8) + 50.625 \text{floor}\left(\frac{i}{8}\right). \quad (3.2)$$

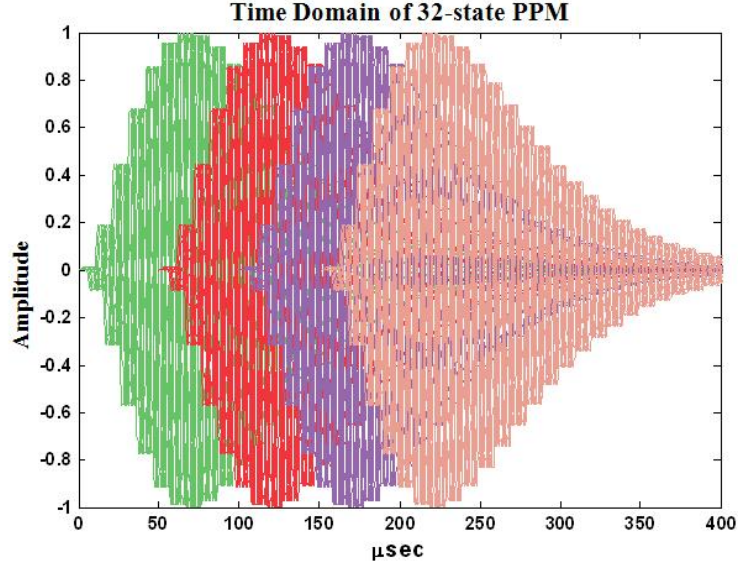


Figure 3.9: PPM 32 on the ninth-pulse

There are two important factors to consider in evaluating demodulation: 1) the signal-to-noise ratio required for data reception, and 2) the skywave and crossrate rejections in a receiver. Even though the skywave and crossrate interference represent the primary source of interference to Loran, only random noise is studied in this thesis. Therefore, the SNR is the primary metric to assess the PPM 32 demodulation performance. Let $y(t)$ be the received signal, and $y(t) = s_i(t) + n(t)$. One technique to demodulate 9th pulse data is a matched filter. A matched filter performs the convolutions of a time-reversed version of a reference signal with the input signal. By multiplying the input signal with a time-shifted version of the reference signal and integrating the product, the maximum of the integrals is the demodulated symbol [29], represented in Figure 3.10.

Assuming that the noise is additive white Gaussian noise (AWGN), the effects of Loran signals in the presence of noise as they pass through the filters are examined. Another assumption is that the filters contribute negligible noise to the signals, so the outputs from each of the filters are correlated; therefore, the noise variance and covariances can be determined. A 30 kHz noise equivalent bandwidth (NEBW) is used in this matched filter model. An upper bound on the probability that a sent

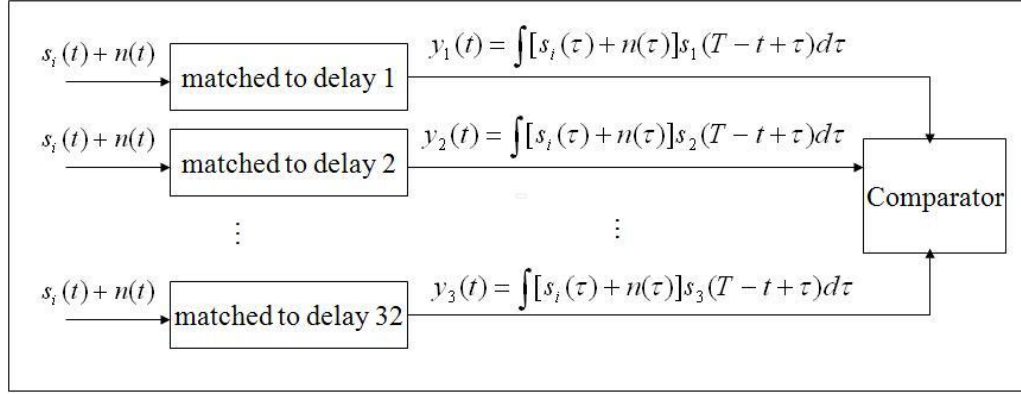


Figure 3.10: PPM 32 matched filter

symbol is not correctly demodulated by a receiver for a given signal-to-noise ratio can be developed. The bound is the sum of the error probability of each incorrectly demodulated symbol in Equation (3.3).

Given the following definitions:

$P(y_i > y_j | j)$ is the probability that the maximum output from the matched filter i is greater than that from the matched filter j , given that the signal j was sent;

F_{norm} is the cumulative density function for the standard normal variable;

d_{ij} is the Euclidean distance between s_i and s_j ; and,

$h(t)$ is a 30 kHz bandpass filter.

Therefore, for PPM 32 ($M = 32$),

$$P_e < \sum_{j=1, j \neq i}^M P(y_j > y_i | i) = \frac{1}{M} \sum_{i=1}^M \sum_{j=1, j \neq i}^M F_{norm} \left(-\frac{\int [s_j(t) - s_i(t)] s_i(t) dt}{\sqrt{\frac{N_0}{2} d_{ij}^2 \int_{-\infty}^{\infty} |h(t)|^2 dt}} \right). \quad (3.3)$$

Figure 3.11 illustrates the error bound for a 32-state PPM as a function of SNR along with simulation results. The discrepancy of the analytic and simulated results likely originates from the use of an ideal bandpass filter for the analytic model and a second-order Butterworth filter for the simulation.

Under the current U.S. proposed ninth-pulse communications (NPC), each Loran

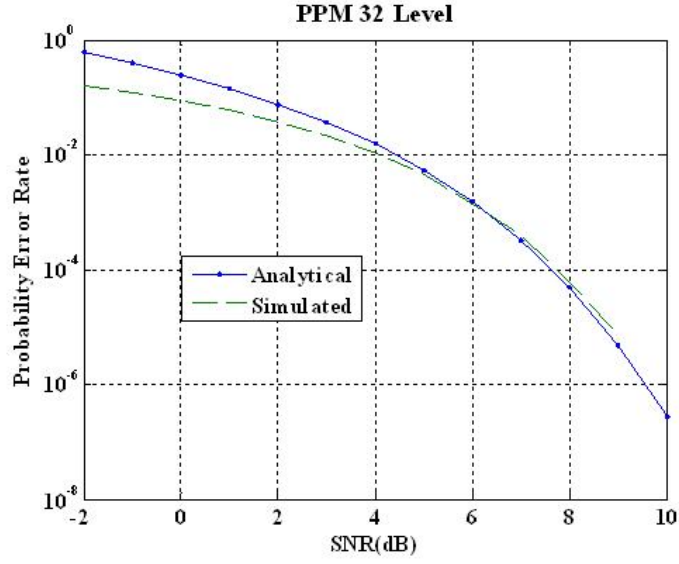


Figure 3.11: 32-state PPM probability of error

message has 120 raw data bits, consisting of a 4-bit header, a 41-bit payload, and 75-bit parity component. The Reed-Solomon (RS) codes are used for the parity check. This forward error correction (FEC) coding method provides an error correction capacity and integrity [31]. It provides the ability to align the message, to correct errors, and to verify that the message has been accurately decoded with high probability. A packet consists of five raw data bits that a modulated pulse can carry in ninth-pulse communications. The packet loss rate can be determined using the overbound for the bit error probability. With 45-bit payload and 75-bit parity check for each Loran message, the percentage of message loss can be calculated using RS coding with an assumption that the packet loss is approximately Gaussian. The performance using RS coding can achieve is:

$$P(\text{error/decoder failure}) = \sum_{j=t+1}^n \binom{n}{j} p^j (1-p)^{n-j}. \quad (3.4)$$

The analytic message loss and packet loss rates are plotted in Figure 3.12. In this figure, the message loss is the probability of decoder failure, and the different packet loss rate originates from the different SNR. Furthermore, it is assumed that packet

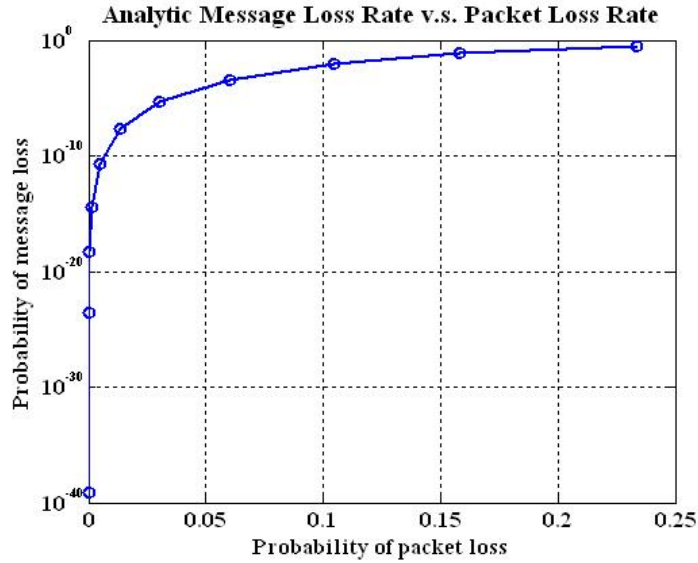


Figure 3.12: Message loss vs. packet loss rate

losses are independent. The message loss rate is essential to estimate the demodulation performance as well as authentication probability, which will be discussed in Section 3.2.

3.2 Integrity Analysis

3.2.1 TESLA Using New Loran Data Channel

The West Coast chain of Loran, GRI 9940, is used to perform the demonstration. Middletown, the closest secondary station to Stanford University, is chosen to implement the authentication scheme to ensure the decoding performance. Figure 3.13 depicts the Loran tower at Wildwood, NJ for illustrative purposes.

TESLA Live Test

At the beginning of 2007, Middletown broadcast both time and authentication messages for testing purposes [35, 41]. The time message was generated by the United



Figure 3.13: Loran transmitter [Picture source: US Coast Guard]

States Coast Guard (USCG) to test the performance of ninth-pulse modulation. Stanford University generated the authentication messages to verify the authentication performance and to demonstrate the geo-security protocol. The time and authentication messages were broadcast in an alternating pattern; each obtained a 50% data capacity. With only one secondary station carrying data messages, a data rate of 50 bits/sec was achieved.

Two different hash functions are required to compute the TESLA one-way chain key values; SHA1 and MD5 were selected for the demonstration. SHA1 outputs a hash value of 160 bits; MD5 outputs a hash value of 128 bits. SHA1 is employed in several widely used security algorithms and protocols. Although MD5 has not been found to be collision-resistant, it has the desired property of one-way-ness. Another reason to use MD5 in this demonstration is its reasonably short digest. A keyed-Hash MAC (HMAC), which is a type of MAC calculated using a hash function in combination with a secret key, was chosen to generate the authentication messages. The hash function used in HMAC was SHA1; as a result, the MAC size was also 160-bit. The key size to create MAC must be at least half of the MAC size to ensure

the security; hence, a 128-bit key was applied. The set of MAC keys were computed using MD5.

A Stanford-generated authentication message consists of a key and a MAC, which result in a total length of 320 bits. With a 41-bit payload in a Loran message, at least eight messages are required to transmit a complete authentication message. Subtypes were used to help the receivers distinguish between the MACs and the keys in the authentication messages. The data type for the authentication message is 0011. Subtypes 1 to 4 represent the identification of MACs; subtypes 6 to 10 are for the transmitted keys. Subtype 5 consists of a 12-bit MAC, a 13-bit padding, and a 12-bit key. A total of ten messages were required to carry one TESLA packet; as a result, it took 23.856 seconds to transmit all of the messages via GRI 9940. The following illustrates the authentication message structure.

```
00110001MAC
00110010MAC
00110011MAC
00110100MAC
00110101MAC
00000000000000Key
00110110Key
00110111Key
00111000Key
00111001Key
```

TESLA uses the one-way key chain and discloses keys in a delayed manner to achieve security. The length of the chain depends on the desired time to the first authentication and the authentication strength. As such, it depends on the available data capacity for authentication. Under TESLA, each segment of the chain consists of a message, a MAC, and the delayed key for a previous MAC. The amount of delay is a design parameter. In the proof of concept demonstration, a three-segment sequence was used. In the demonstration, half of the ninth pulse data capacity was used for authentication messages. A time message and authentication messages were sent alternately. To simplify the implementation, three segments were generated and

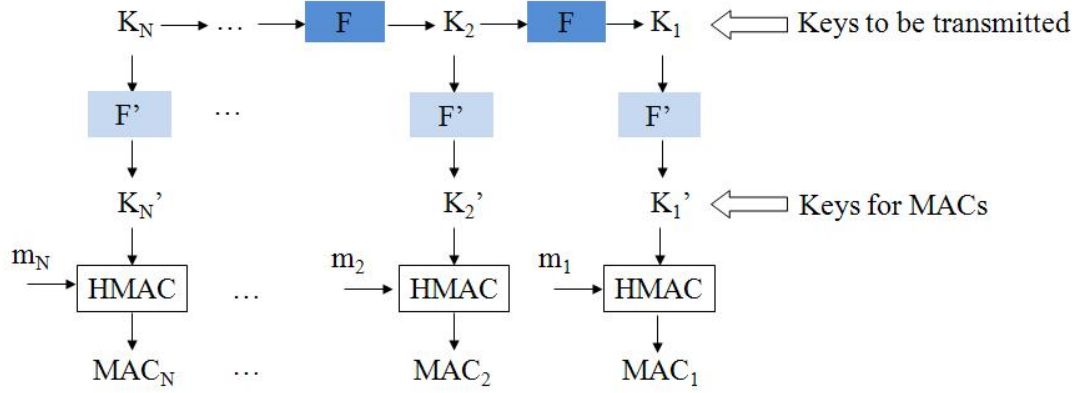


Figure 3.14: Stanford generated Key and MAC for TESLA demonstration

broadcast repetitively. In the setup phase, K_3 was randomly generated, and the transmitted key chain (K_2, K_1) was computed using SHA1 based on K_3 , that is, $K_2 = F(K_3)$, $K_1 = F(K_2)$, and $F = \text{SHA1}$. MD5 was used to generate the keys for MAC generation, $F' = \text{MD5}$. These MAC generation keys were represented as $K'_1 = F'(K_1)$, $K'_2 = F'(K_2)$ and $K'_3 = F'(K_3)$, which were used with the messages m_1, m_2 and m_3 to compute MACs, $h_1 = \text{HMAC}(K'_1, m_1)$, $h_2 = \text{HMAC}(K'_2, m_2)$, and $h_3 = \text{HMAC}(K'_3, m_3)$, respectively. Figure 3.14, a simplified version of Figure 2.5, represents the roles of the hash functions, the MAC function used, and the computations of the three segments.

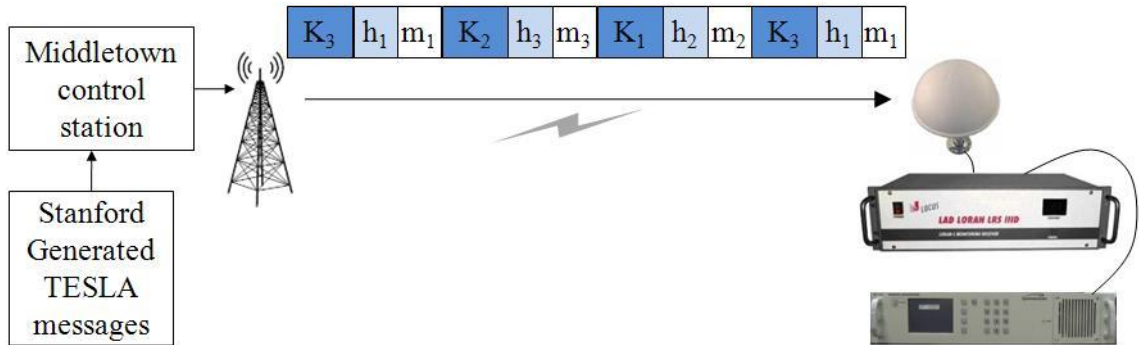


Figure 3.15: Circular TESLA chain on Middletown

In the broadcast phase, the three segments are transmitted in a sequence of $\langle m_1, h_1, K_3 \rangle$, $\langle m_2, h_2, K_1 \rangle$ and $\langle m_3, h_3, K_2 \rangle$. Such a message sequence is illustrated

in Figure 3.15.

K_1 , the first key of generation and last transmission key, is supposed to be embedded in the receiver. Once enough segments are received and buffered, the three verification steps are performed.

1. First stage key verification: Compare the received K_1 with the embedded key. If they are the same, move on to the next step. This step verifies that the source is the same as the one that provided the key.

2. Second stage key verification: Hash the received keys using SHA1 and compare them with the keys in the previous packet. This step verifies authenticity of the source. That is, another signal source has not been injected.

$$\text{SHA1}(K_2) \stackrel{?}{=} K_1$$

$$\text{SHA1}(K_3) \stackrel{?}{=} K_2$$

3. MAC verification: Construct the MAC keys using MD5 and compute the MACs with these keys and the received messages. Compare these computed MACs with the received ones (h_1, h_2, h_3). The signal is validated if they all match. This verifies that the message has not been altered.

$$\text{HMAC}(\text{MD5}(K_1), m_1) \stackrel{?}{=} h_1$$

$$\text{HMAC}(\text{MD5}(K_2), m_2) \stackrel{?}{=} h_2$$

$$\text{HMAC}(\text{MD5}(K_3), m_3) \stackrel{?}{=} h_3$$

Authentication Performance Evaluation

The performance of TESLA depends on the signal-to-noise ratio, the demodulation performance, and authentication data capacity. A matched filter model in the presence of noise for the receiver processing of the signal is applied to analyze the bit error rate. Additive white Gaussian noise (AWGN) is assumed to pass through the filter. The noise variances are used to determine an upper bound on the error probability, which is the probability of a sent symbol that is not correctly received by the receiver, for a different SNR. The received SNR is determined by the range from a transmitter to a receiver, the transmitter radiated power, and the local noise level. The field strength of the Loran groundwave, modeled with a surface with finite conductivity, is provided as

$$\begin{aligned}
E_{sig}^2 &= \left(\frac{9.48}{1000r}\right)^2 \cdot P \cdot 10^{-0.1a\left(\frac{r}{1000}\right)^b}, \\
a &= 17.52, \\
b &= 1.1036,
\end{aligned} \tag{3.5}$$

where r is the range from a transmitter to a receiver in km, and P is the transmitter-radiated power in watts [29]. The terms, a and b , are derived from a least square curve fit, and depend on land conductivity and transmitted frequency. The attenuation factor, $10^{-0.1a\left(\frac{r}{1000}\right)^b}$, increases from a seawater path to a nonhomogenous land path. A constant noise level, 52.4 dBu, is assumed for the GRI 9940 coverage area [3]. The predicted atmospheric noise field strength is estimated for a judiciously selected point in the middle of the service area of the West Coast chain. The predicted noise is the average of noise levels that exceed 5% of the time over four-hour period of a day for each season of the year. The Middletown contour of analytical signal strengths is plotted in Figure 3.16.

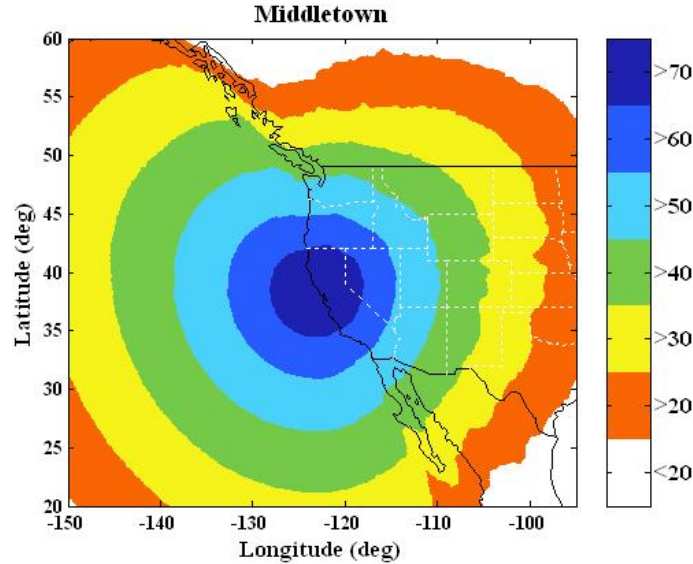


Figure 3.16: Middletown signal field strength contour plot, signal strength in $\mu\text{V}/\text{m}$

One GRI can carry five symbols, which is considered to be a packet. Once the

probability of symbol error is determined, the average and standard deviation of the packet loss rate can be estimated. Assuming that the packet loss distribution is approximately Gaussian, the message loss, indicated in Figure 3.12, can be calculated using FEC [29]. Based on the number of Loran messages needed to carry one TESLA segment, the probability of authenticating or verifying a TESLA segment correctly yields

$$\begin{aligned} P_{authentication} &= (1 - p)^N \\ N &= \text{ceil}\left(\frac{10}{\rho_{data}}\right), \end{aligned} \quad (3.6)$$

where N is the sum of the number of Loran data messages to authenticate or the number of Loran messages to carry one authentication message, and p is the message loss rate shown in Figure 3.12. ρ_{data} is the fraction of data capacity for authentication, which refers to the percentage of messages whose sole purpose is to authenticate data. For instance, GRI 9940 has a raw data rate of 50 bits/sec. A 50% data capacity results in an authentication raw data rate of 25 bits/sec. The number of Loran messages carrying one authentication message is fixed. Hence, as the ρ_{data} decreases, the number of data messages required to authenticate increases, resulting in an increase in N . Applying Equations (3.5) and (3.6), a contour plot is developed to analyze the authentication probability geographically.

As the SNR increases, the symbol error probability decreases, which leads to a decrease in the message loss rate and an increase in the authentication probability. For each SNR, the message loss probability is fixed. Depending upon the implementation of TESLA, the available data capacity for authentication messages determines the number of Loran messages required to carry the data messages, the keys, and the MACs. As mentioned in Subsection 3.1.3, each Loran message consists of 120-bit symbols with a payload of 41 bits. Therefore, an increase in the fractional data capacity for authentication results in a decrease in the number of Loran messages carrying each TESLA segment. Assuming that each Loran message is broadcast independently, the probability of authentication can be calculated, which is also proportional to the data

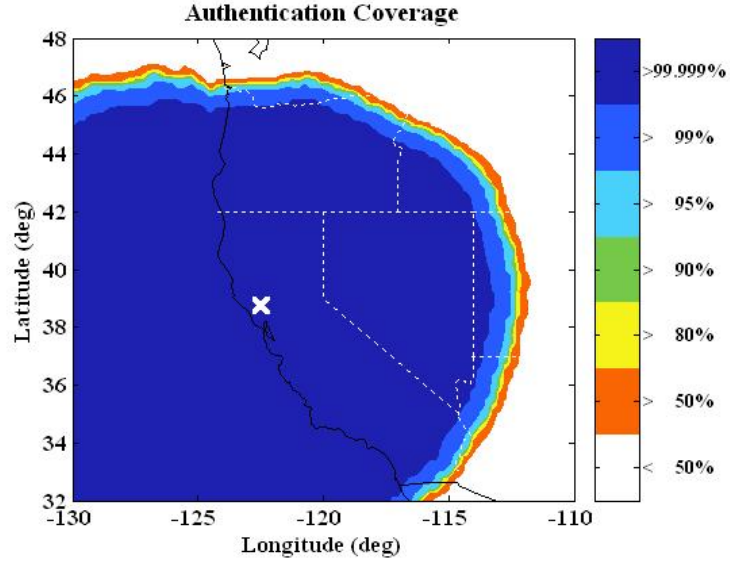


Figure 3.17: Middletown authentication probability as a function of user locations

capacity. Another important parameter to test the performance of TESLA is the authentication time, or time-to-alert (TTA), which is the average time that is required before a user can perform an authentication. Similar to the probability analysis, Figure 3.18, obtained from Equation (3.7) illustrates the authentication time that is controlled by the received SNR and the authentication data capacity.

$$ToA = \frac{a \cdot (N_{authentication} + N_{data}) \cdot \frac{m}{r}}{(1 - p)^{N_{authentication} + N_{data}}} \text{ sec} \quad (3.7)$$

where $N_{authentication}$ is the number of Loran messages for authentication, N_{data} is the number of Loran data messages, m is the number of data bits in one message ($m = 120$), r is the data rate ($r = 50$ bits/sec for Middletown), p is the message loss rate, and a is a scale factor to guarantee the entire authentication messages are received ($a \geq 1$).

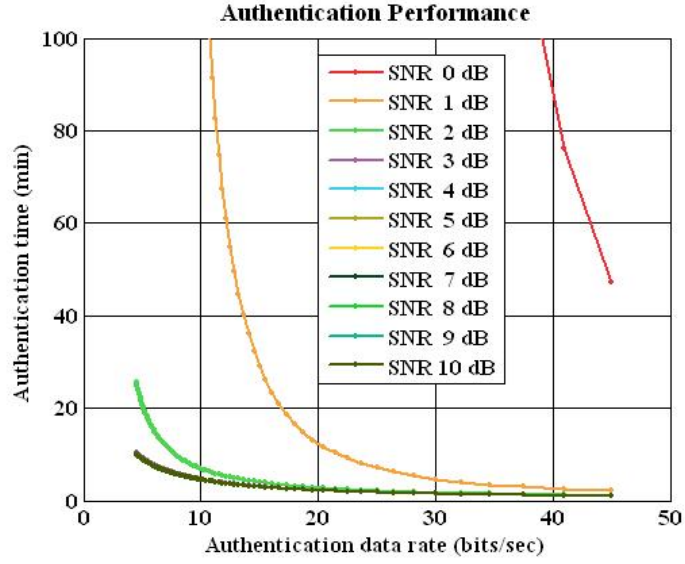


Figure 3.18: Authentication time is inversely proportional to data capacity and SNR.

To test the performance of TESLA, a data collection trip from Stanford to Los Angeles, CA was made. Some of the data collection equipment is shown on the left of Figure 3.19. An E-field Loran Locus antenna was connected to a Locus LRS IIID receiver to capture RF signals. The receiver only functioned as a front-end to amplify and filter the incoming RF signals. The output of the Locus receiver went into the Enhanced Loran Research Receiver (ELRR) to digitize and process the conditioned signals. ELRR also decoded the data bits modulated on the 9th pulse of Middletown. A serial port was used to allow MATLAB to log data from ELRR. The white dots plotted in the contour plot of Figure 3.19 represent the test locations. The results illustrated that the Middletown signal source was successfully authenticated at all of these locations. With a 50% data capacity for the authentication messages and the current LDC proposal, the average time for a receiver to authenticate a Loran station is approximately 38.4 seconds. This authentication time will be extensively used in Subsection 3.2.4 to quantify the geo-security integrity and the location-based attacks.

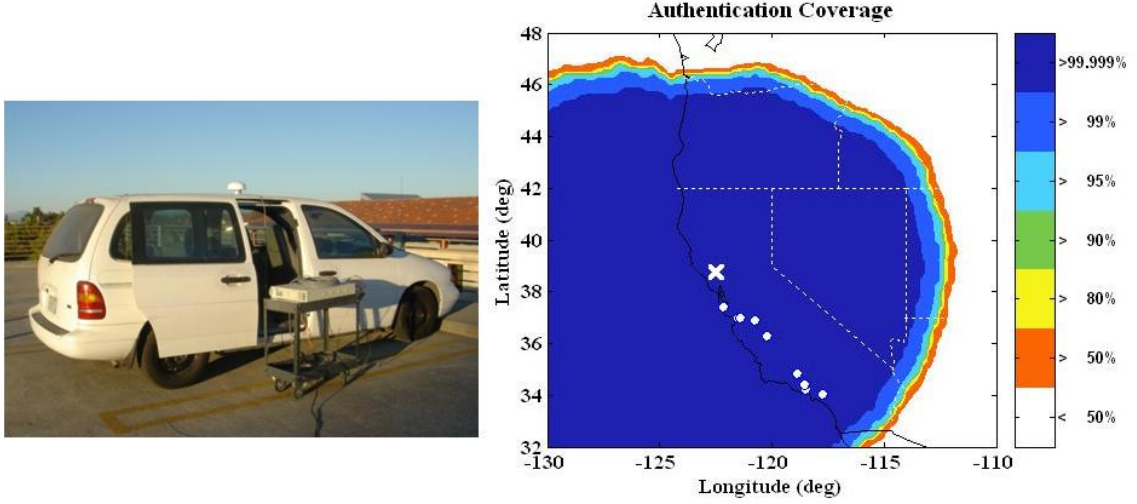


Figure 3.19: Data collection setup and the test locations

3.2.2 Parameter Spatial Decorrelation

Spatial decorrelation is a measure of uniqueness for location-dependent parameters. It is the change or rate change of the location-dependent parameters as a function of physical locations or distances. High spatial decorrelation indicates that users can be distinguished from each other with a small separation. The false accept rate is used to characterize and quantify spatial decorrelation.

To examine the spatial decorrelation of location-dependent parameters, a data set was collected in a parking structure at Stanford University. Eleven test locations were chosen with a separation of 3 meters, depicted in Figure 3.20. Test Location 1 is considered to be the master location. In other words, it is the location of the authorized user. The spatial decorrelation of location-dependent parameters is analyzed for various separations between the master location and the rest of the test locations.

Applying the P_{FA} estimation in Equation (2.24), the spatial decorrelation of different stations in GRI 9940 for one particular location-dependent parameter, TD, was first compared. The comparison result is illustrated on the left-hand side of Figure 3.21. The x -axis represents the separation between the test locations from the master location in meters, while the y -axis provides the estimated false accept rate values.



Figure 3.20: Test points in a parking structure at Stanford University

Middletown, the closest station to Stanford campus, is approximately 153 km away; hence, its SNR is the highest due to the shorter propagation distance. The quantization steps were estimated based upon the received SNR of the different stations, 3 meters for Middletown and 15 meters for George and Searchlight. The Middletown FAR, plotted by the red curve, decorrelates faster compared with that of George and Searchlight. Therefore, the received SNR has a significant impact on the parameter spatial decorrelation. A smaller parameter quantization step can be applied when the received SNR is high, thus resulting in a better spatial decorrelation.

To study the uniqueness or the strength of different location-dependent parameters, the spatial decorrelation was compared using the measurements from the Middletown station. As illustrated in Figure 3.21 on the right, TD has the strongest spatial decorrelation, whereas the SNR is the least spatially decorrelated. Generally speaking, the SNR is sensitive to the environment and the local noise; its small spatial variation in this experiment is primarily caused by the open-sky environment and closely-separated test locations.

A decorrelation distance refers to the minimum distance from the master location at which the FAR value is less than a reasonably small threshold. The threshold

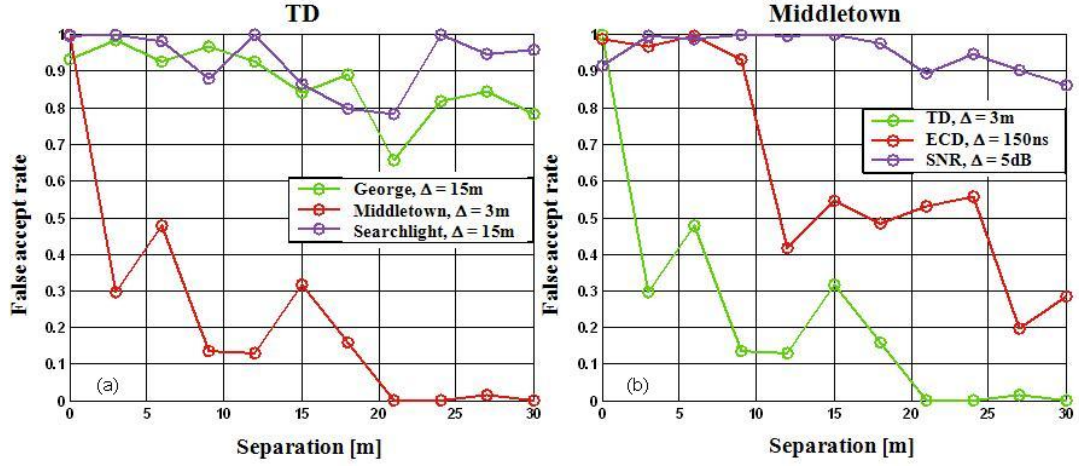


Figure 3.21: Spatial decorrelation of Loran location-dependent parameters. Quantization steps were chosen based on the received SNR. (a) TD for different stations: $\Delta = 15\text{ m}$ for George; $\Delta = 3\text{ m}$ for Middletown; $\Delta = 15\text{ m}$ for Searchlight. (b) Parameters from Middletown: $\Delta = 3\text{ m}$ for TD; $\Delta = 150\text{ ns}$ for ECD; $\Delta = 5\text{ dB}$ for SNR.

is chosen to be 0.01. With this threshold, a decorrelation distance is computed by curve fitting the estimated FAR values with an exponential function. A fitted curve is plotted by the dashed line in Figure 3.22. The decorrelation distance for TD from Middletown in this particular experiment is approximately 18 meters, that is, attackers who are 18 meters away from the master location would result in a FAR of less than 0.01. A decorrelation distance can be used as a guide to choose the appropriate quantization step for geotag generation.

Choosing an Optimal Quantization Step (Δ)

An optimal quantization step selection should take into account both the geotag reproducibility (FRR), and the spatial decorrelation and system security (FAR). A diagram showing how to specify an appropriate step size with which the system performance can be optimized is illustrated in Figure 3.23. The desired FRR and FAR are the design parameters and specified by users. A quantization step is estimated based upon the received signal condition and the desired FRR, as discussed in Section 2.3. A spatial decorrelation test is conducted to compute the decorrelation distance

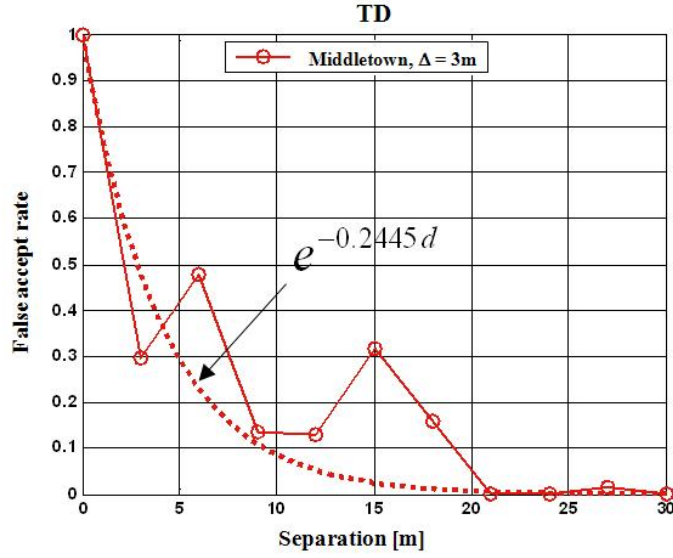


Figure 3.22: Spatial decorrelation of TD: measurements and curve fitting

associated with the desired FAR. If the decorrelation distance is less than the physical security radius, the system can meet the requirements of both reliability and security. On the other hand, if the decorrelation distance is greater than the physical security radius, the quantization step should be adjusted to balance the tradeoff between FRR and FAR. For instance, the step size should be reduced to aim for the desired FAR whereas the step size can remain the same to satisfy the geotag reproducibility requirement.

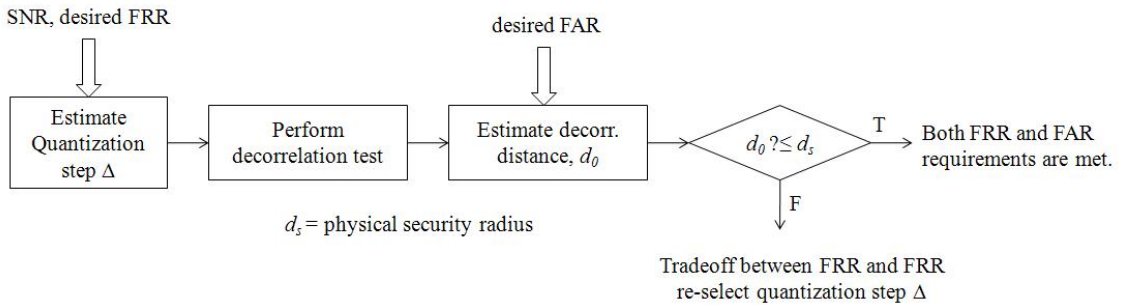


Figure 3.23: A process to choose an optimal quantization step

Spatial Decorrelation Comparison Using Different Geotag Generations

The same parking structure data set is applied to compare the spatial decorrelation of different geotag generation algorithms [40]. The FARs of three geotag generation algorithms – support vector machines (SVM) classifier-based, k-nearest neighbor (kNN) classifier-based, and quantization-based – are estimated and illustrated in Figure 3.24. A thorough discussion of classifier-based geotag generation algorithms is given in Appendix B with examples.

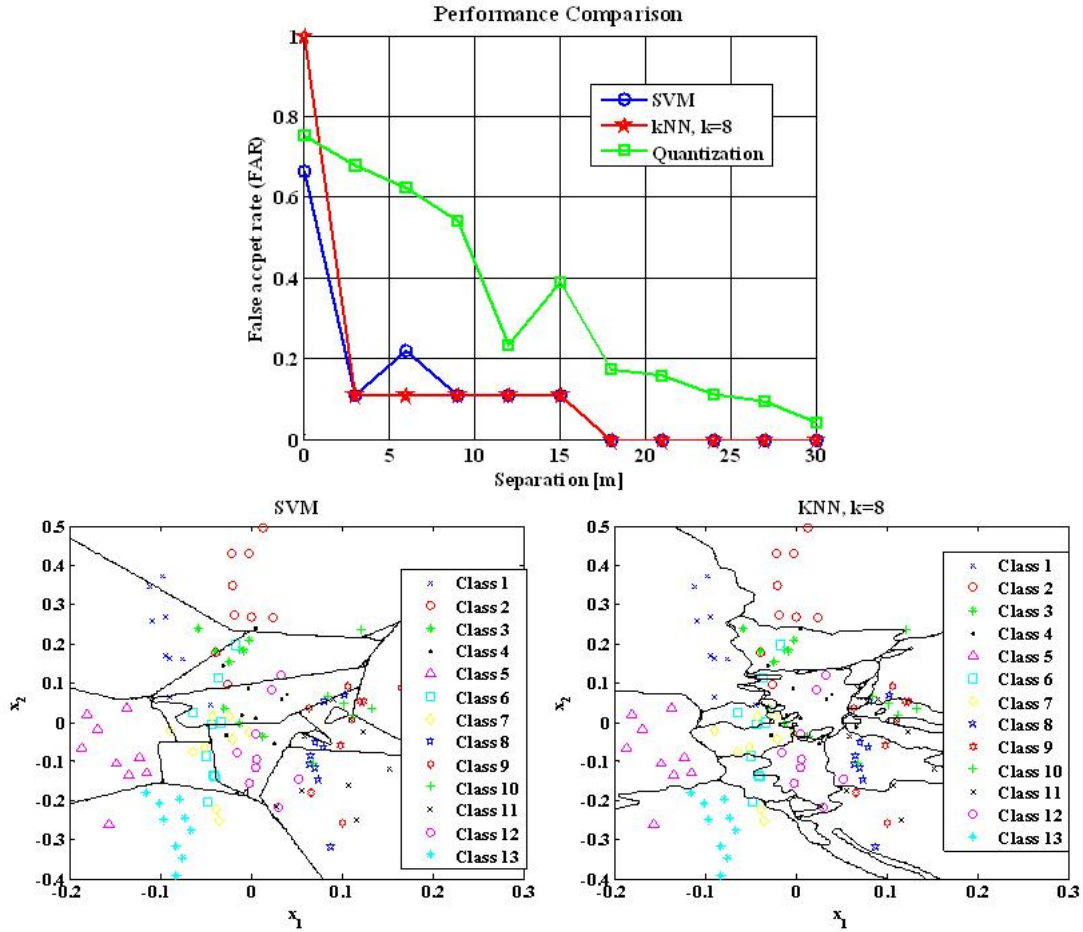


Figure 3.24: Spatial decorrelation comparison of quantization-based, kNN and SVM classifier-based geotag generation algorithms

The 11 location-dependent parameters (TD, ECD, and SNR from four West Coast stations) are the inputs to the geotag generation algorithms. For the classifier-based

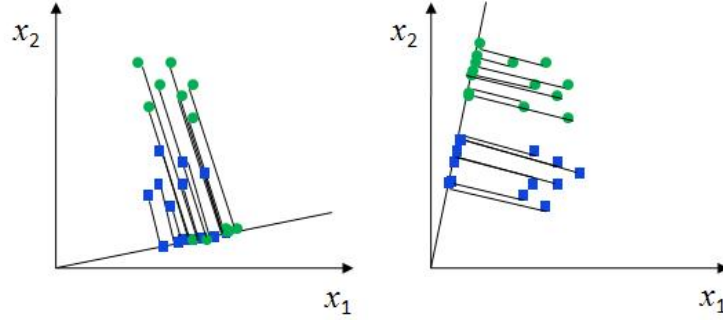


Figure 3.25: Linear dimensionality reduction from two-dimensional to one-dimensional: examples of bad projection (left) and good projection (right)

geotags, a linear dimensionality reduction algorithm is applied to lower the input vector dimension to two to achieve better spatial discriminations. The algorithm processes the vector of measurement of signal characteristics by projecting the vector into a lower dimension. Figure 3.25 illustrates the linear projection of two-dimensional data samples into one-dimensional.

The bottom plots in Figure 3.24 indicate the classification visualization of the two classifier-based geotags. The kNN classifier produces smoother decision boundaries compared to the SVM-based. The comparison result shown on the top of Figure 3.24 indicates that the kNN classifier-based geotag has the best spatial decorrelation, whereas the quantization-based geotag has the worst. For instance, at a separation of 3 meters, the FAR is reduced by 83.7% from 0.68 of the quantization-based geotag to 0.11 of the classifier-based geotag.

3.2.3 Resolution of Loran Geotag

While false accept rates measure the spatial decorrelation of location-dependent parameters in the continuous domain, the geotag resolution is essential to the positioning accuracy in the discrete domain, which can be important to many geotag applications, such as Loopt and digital manners policy. Three different sites were selected to perform the resolution test: a parking structure, a soccer field, and an office building. Spatial data were collected for 5 minutes at the test locations at each site. An H-field

antenna and Locus SatMate receiver, shown in Figure 3.26, were used for the data collection. The Locus SatMate receiver averages and outputs Loran location-dependent parameters every minute.

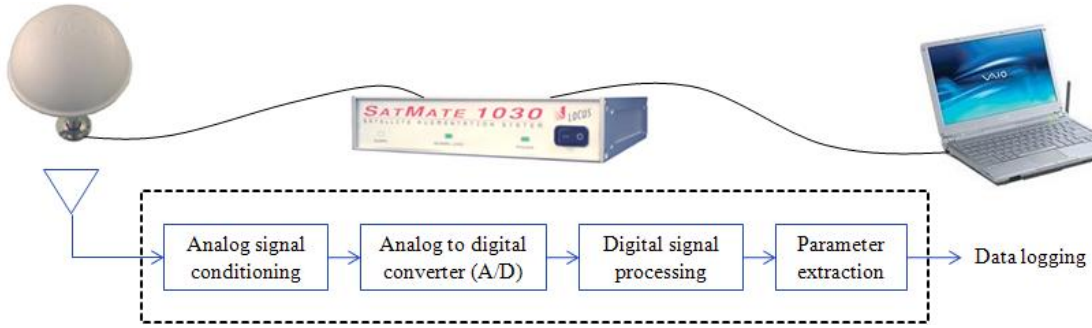


Figure 3.26: Loran data collection setup: Locus H-field antenna, Locus SatMate 1030 receiver, and laptop for data logging.

- **Site 1.** The first data set was collected at 21 different test points on the top floor of a parking structure at Stanford University. This place has an open sky view and no obstructions from the environment, but there are some metal structures nearby. The altitude is relatively high compared with the other two sites. The dimensions of the parking structure are approximately 70×50 meters.
- **Site 2.** The second data set selected 16 test points on Roble field, a soccer field at Stanford University. This environment has some obstructions from the trees and buildings close by. The field has the dimensions of 176×70 meters, so the distribution of the test locations is less dense compared to the other two sites.
- **Site 3.** The third data set, containing 21 test points, was collected on the top floor, both inside and outside the fourth story, of the Durand office building. The concrete building with metal frames significantly attenuates signal strength but introduces more spatial decorrelation in the location-dependent parameters, which is beneficial in computing geotags.

The triple (TD, ECD, SNR) data set from four stations in the West Coast chain, GRI 9940, were utilized to construct geotags. Quantization steps are estimated based on

the measured SNR. Low SNR signals are often attenuated more and pick up more noise. Thus, parameters from low SNR stations are less consistent. Therefore, the quantization steps were selected based on the received SNRs as discussed in Figure 3.21. The two-dimensional cells were created using Voronoi diagrams, and colors were mapped into the cells based on the computed geotags accordingly. The color map was superimposed on a Google map. A color bar, shown at the bottom of the superimposed pictures, represents the unique geotags. These color cells can help visualize the geotag resolution in a two-dimensional view. Each black dot at the center of the cells indicates a test location.

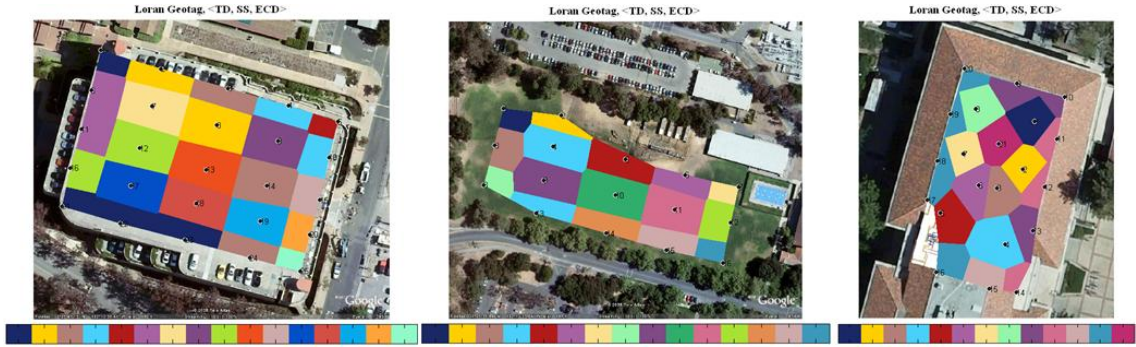


Figure 3.27: Geotag 2-D visualization: parking structure (left); soccer field (middle); office building (right). The cells are created using Voronoi diagrams; each color represents a different geotag computed from measured location-dependent parameters.

The left picture in Figure 3.27 represents the geotag plot on the top floor of the parking structure; the middle plot is the soccer field; the right plot shows the top floor/roof of the office building. Loran signals are sensitive to the environment, especially to large metal structures. The re-radiation of signals from metal causes more distortions in the RF signals, thus resulting in more randomness and higher resolution of geotags at certain locations. Such scenarios were observed from the three pictures in Figure 3.27. The test locations with small separations still result in unique geotags. It is worth mentioning that only two stations, Fallon and Middletown, were used to compute the geotags for Site 3, whereas the other two sites used all four stations from GRI 9940. Due to the low signal strength indoors, the SatMate receiver was not able to acquire the two low SNR stations, George and Searchlight, inside the

office building. The averaged resolution of the three different sites is as follows:

- The 2-D cell diameter in the parking structure ranges from 8 meters to 35 meters. There are four locations that result in the same geotag shown in dark blue on the left-hand side of Figure 3.27.
- The resolution of geotags in the soccer field is lower compared with that of the parking structure due to the large separations between the test locations. The average size of the colored cells that represent geotags is approximately 30×50 meters.
- The smallest colored cell or the highest geotag resolution is approximately 5 meters depicted in purple in the right plot of Figure 3.27. A range on the actual geotag resolution at this location is 8×20 meters.

3.2.4 Location-Based Attacks

This section evaluates and quantifies the location-based attacks using real Loran data. To bypass the location validation or achieve an authentic geotag, attackers must be physically close to a legitimate user's location because a tamper-resistant device and self-authenticated signals preclude off-site spoofing attacks. The first experiment in this section was conducted to examine the location-based attacks by replicating a real attack scenario. The data was collected within the same office building in which the geotag resolution test was performed, pictured in Figure 3.28. The center point marked "User" is considered to be a legitimate user's location; the other four markers represent the attackers. Assuming that there is some degree of physical security, an attacker cannot be at the legitimate user's location, but he can access the nearby locations. Thus, these kind of attacks have been called "parking lot" attacks.

The parameter distance of TD, SS, and SNR between the user and the attackers is plotted in Figure 3.29. The x -axis represents the separation between the user and attackers in meters: 38.16 for Attacker 1, 77.14 for Attacker 2, 20.24 for Attacker 3, and 51.95 for Attacker 4. The upper left plot illustrates the TD measurements from George, Middletown, and Searchlight. It is observed that Attackers 2 and 4 have smaller parameter distance with respect to the user, whereas Attackers 1 and 3 have



Figure 3.28: Location-based attack data site

a larger distance. This uncertainty, introduced by the environment, such as trees and buildings, increases the unpredictability of a geotag. For instance, although Attacker 2 is farther away from the target location, less attenuation of the signal results in a smaller absolute distance in the received TD. The lower left plot represents the distance of the signal strength between the user and the attackers. The user's antenna was placed on the roof of the office building; thus, the received signal strength is higher. Such measurements explain why most of the numbers in the middle plot are negative. The lower right plot illustrates the SNR of all four stations, which does not follow the exact trend of the signal strength, because the noise floor contributes randomness to the signals. Weak correlations between the location-dependent parameters are observed; thus, this dissertation ignores the correlation component when analyzing the integrity and continuity of a geo-security system. The correlation between the parameters is worth to be considered for future research.

The data set was first applied to quantify a location brute force attack or the simple attack, defined as follows: the attackers do not instigate any kind of organized spoofing attack, but merely rely on measurement errors to achieve the user's geotag by accident using their received location-dependent parameters. The false accept rate, P_{FA} , is used to quantify the difficulty of this attack. The number of trials that an attacker needs to search for a correct geotag is approximated as $N_{trials} = \frac{1}{P_{FA}}$.

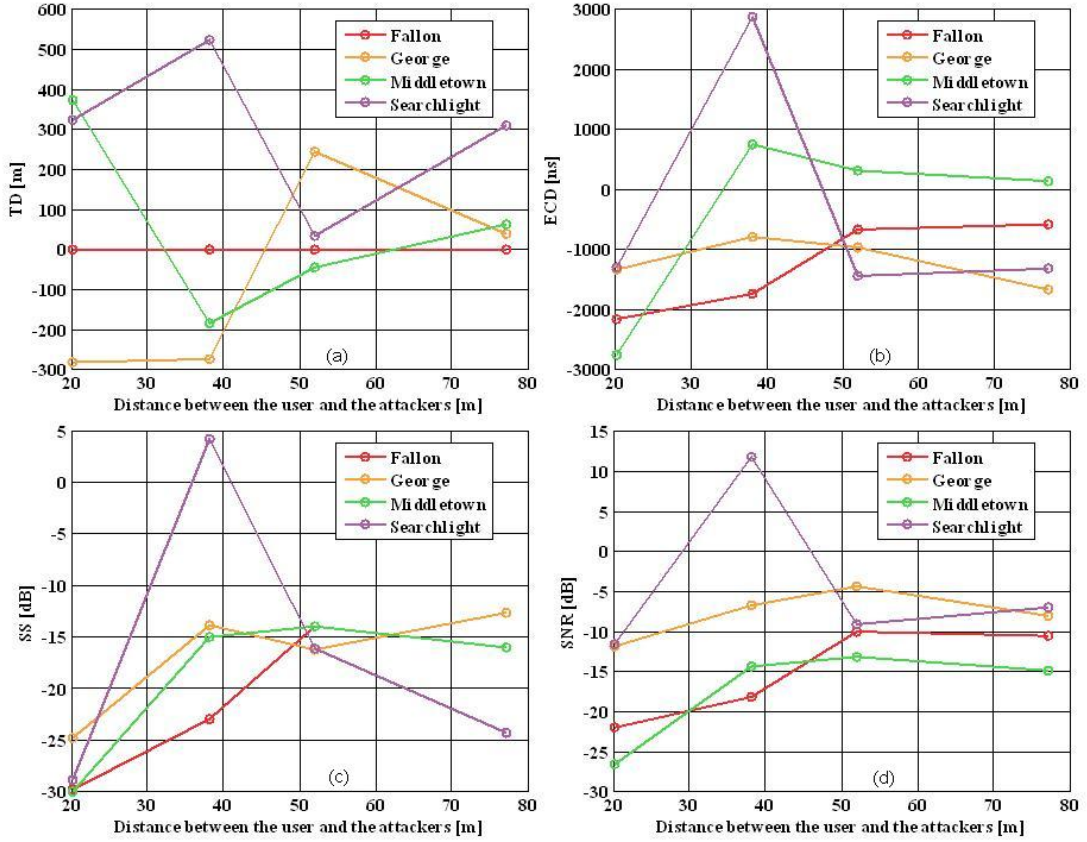
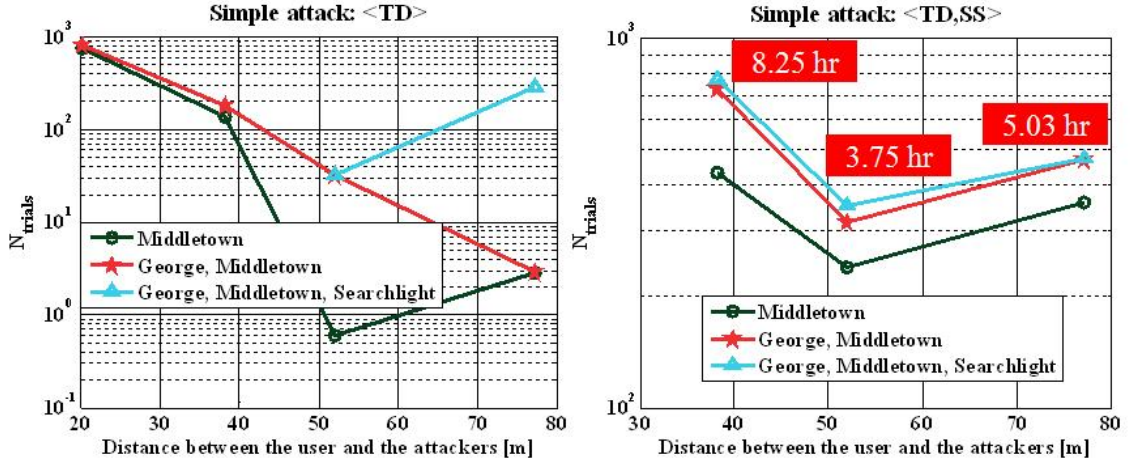


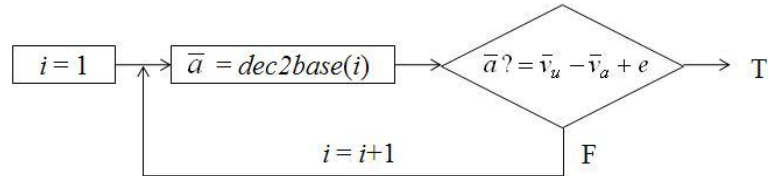
Figure 3.29: The parameter distance between the user and the attackers as a function of physical distance: (a) TD; (b) ECD; (c) Signal strength; (d) SNR.

Assuming that the location-dependent parameters are uncorrelated, the N_{trials} estimated from the measurements is presented in Figure 3.30. The left figure illustrates the performance using one location-dependent parameter, TD, from George, Middletown, and Searchlight. The results indicate that using only one parameter is not sufficient to provide good performance. By adding one more parameter—signal strength—the improved N_{trials} is plotted on the right. The data point for Attacker 3 (separation = 20.24 m) is missing because its N_{trials} is too high, almost to infinity. Taking into account the fact that each trial takes 38.4 seconds due to the signal source authentication, the total attack time is estimated and depicted in Figure 3.30.

The same data set was employed to estimate the N_{trials} in the selective delay

Figure 3.30: N_{trials} for the location-based simple attack

attack, analyzed in Subsection 2.2.4.2. Equation (2.25) is applied to estimate an upper bound and a lower bound on the number of guesses required to identify the correct geotag at the four attackers' locations. An upper bound indicates that an attacker has little knowledge of the signal characteristics and must go through an exhaustive search, in which case the weighting vector is $w = [1, 1, \dots, 1]^T$; the lower bound is the case in which the attackers correctly predict the location-dependent parameters differences, and w is a unit vector. The exhaustive or organized search is discussed as follows and shown below in Figure 3.31:

Figure 3.31: The exhaustive search algorithm for the selective delay attack. \bar{v}_u is the user's discrete parameter vector, and \bar{v}_a is the attacker's discrete parameter vector.

1. Let \bar{v}_u be the discrete parameter vector of the user, \bar{v}_a that of the attacker, and $\bar{v} = \bar{v}_u - \bar{v}_a + e$, where e is an offset chosen to guarantee that the elements in \bar{v} are non-negative.
2. Initialize $i = 1$.

3. Compute $\bar{a} = \text{dec2base}(i)$.
4. Compare the elements in \bar{a} and \bar{v} . If they are the same, the changes that the attacker must make in his discrete location-dependent parameters equal $\bar{a} - e$; otherwise, increment i , such that $i = i + 1$, and loop through Steps 3 and 4.

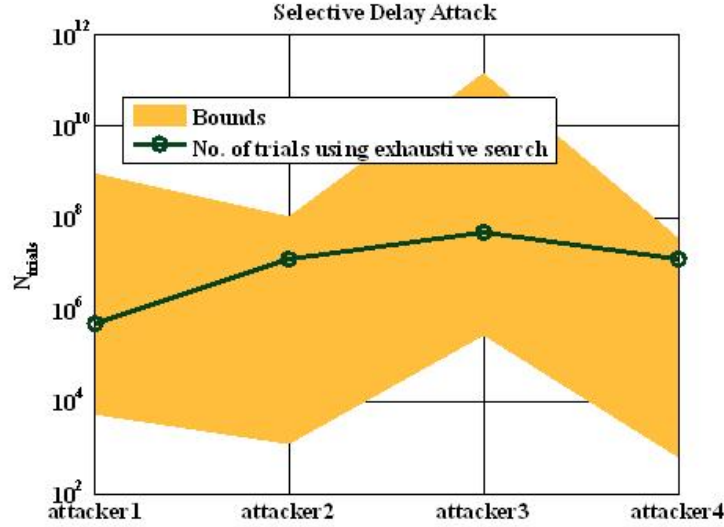


Figure 3.32: N_{trials} for the selective delay attack

Figure 3.32 illustrates the bounds and the number of trials needed to search for the target geotag using an exhaustive search. The results prove that a selective delay attack is not a threat to geo-security systems in this particular scenario. Even in the worst-case scenario, Attacker 4 has a lower bound on N_{trial} of 1024. Considering that each trial takes 38.4 seconds due to signal authentication, it would take the attacker 10.9 hours to crack the system. Unlike the conventional cryptographic attacks, the selective delay attack requires the attackers to be physically close to a target user's location. Attackers search through the different combinations of parameter modifications manually by processing and altering the analog signal characteristics. Such an attack takes time and effort.

The number of trials needed to guarantee success with the organized search is not linearly dependent on the physical distance between the target user and the attackers.

For instance, Attacker 2 is farther from the target location than Attacker 1; however, his N_{trial} bounds are lower than those of Attacker 1. The obstructions, such as trees and buildings, insert more uncertainties and randomnesses into the computed geotag, raising the security level of the system and making the attack more difficult.

Two solutions can possibly improve the security level of geo-security systems. First, integrate discrete time information into geotag generation. The time information source should be: 1) independent of the spatial component, and 2) difficult to spoof. The Loran data channel broadcasts time messages, which can be blended with the location-dependent parameters in geotag generation. As such, attackers cannot indefinitely try different combinations to perform the exhaustive search for the target geotag. The second option is to increase the spatial decorrelation of a derived geotag by using more location-dependent parameters and transmitters. Short-range signals, such as Wi-Fi, Bluetooth, RFID, or UWB, can be used with Loran to expand the security radius. A combination of different signals increases not only the spatial decorrelation, but also the signal variation and the information entropy. In addition, the security radius can be increased by improving the physical security at a user's location, such as security guards or fences.

3.3 Continuity Analysis

Continuity risk refers to the probability that authentic users fail to validate their locations or experience loss of services during all hours of operation of geo-security services. Both the transmitter side and receiver side can affect the continuity risks. A signal transmitter is not always broadcasting and may be shut down due to maintenance or other implementation issues. On the other hand, there are different error sources on the receiver side, such as random noise, seasonal bias, and receiver problems. As a result, if a quantization step is not selected properly, there is a likelihood that the computed geotag is not reproducible.

3.3.1 Seasonal Monitor Data

This section evaluates the temporal variations of Loran data using seasonal monitor data. One of the error sources in Loran signals is the additional secondary factor (ASF), that is, the extra delay in propagation time due to the signals travelling over a mixed path: partially over land with various conductivities, terrain surfaces, and elevations, and partially over seawater. This delay is significant and can introduce a position error of hundreds of meters [30]. ASF represents one of the largest error sources in Loran. Many Loran researchers have been monitoring and studying its characteristics in order to model its seasonal variation and provide an overbound error for Loran users. To observe this seasonal variation, data with a long time span should be collected.

A seasonal monitor station, equipment provided by Alion Science & Technology, shown in Figure 3.33, has been set up at Stanford University to study the ASF characteristics on the West Coast. A Locus E-field antenna and a Locus SatMate 1030 receiver are used to continuously log Loran location-dependent parameters. A GPS receiver is used to train the Loran receiver clock. The surveyed GPS antenna position is taken as a reference for ASF corrections. The Loran receiver averages the parameters every minute.

A 90-day data set was used to investigate the temporal variations. The raw TOA data with zero mean, ECD, and SNR from the Loran West Coast chain are plotted in Figure 3.34. Loran West Coast chain, GRI 9940, includes four stations: Fallon, George, Middletown, and Searchlight. The ASF seasonal variation is observed in the TOA plot on the left.

The evaluation of consistency was conducted using the TOA measurements from Middletown. Due to the seasonal change of ASF, the standard deviation of the measurements increases with time; as a result, the data do not follow a Gaussian distribution. The histogram of the TOA measurements with zero mean is illustrated in Figure 3.35. The standard deviation for the 90-day measurements is approximately 12.19 meters. The red curve represents the Gaussian distribution constructed using the measured standard deviation. To measure temporal entropy accurately and build a robust geo-security system, it is necessary to first remove this seasonal-varying bias.

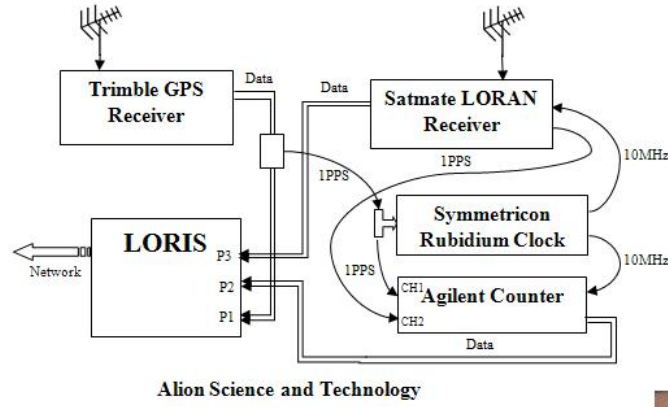


Figure 3.33: Stanford seasonal monitor station: a block diagram of connections and data collection equipment

Otherwise, it would significantly increase the continuity risk for the authorized user of location security. Many factors affect ASF, including the conductivity of the soil, temperature, humidity, local weather, etc. Therefore, ASF varies both temporally and spatially, which raises the difficulty in modeling ASF over CONUS. The temporal component originates from all time-varying aspects, whereas the spatial component takes into account the non-uniform ground conductivity and topography [55]. From previous studies and observations of seasonal monitor data, winter has the most significant variations. Furthermore, the East coast has significantly greater variations than the West Coast.

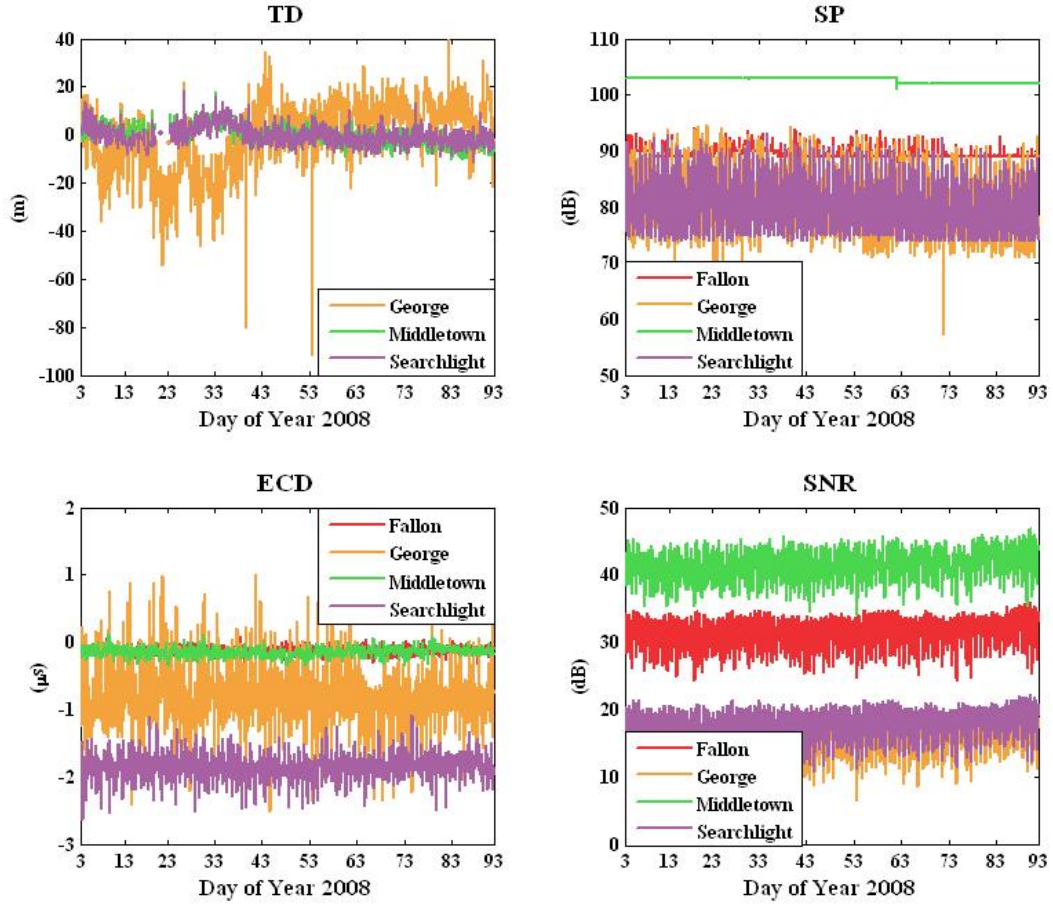


Figure 3.34: Loran 90-day seasonal monitor data from the West Coast stations: TD, signal strength at peak, ECD and SNR.

3.3.2 ASF Mitigation

Since ASF significantly contaminates Loran signals and increases the continuity risks, it is important to mitigate the seasonal bias before the geotag generation. In addition, ASF mitigation improves the security strength by reducing the false accept rate of TOA as both the users' and attackers' TOA standard deviations are lowered and a smaller TOA quantization step can be applied. The presence of ASF in Loran signals degrades the geotag reproducibility and increases the system continuity risks. Thus, it is necessary to develop ASF mitigation techniques to correct the seasonal bias before computing geotags. Many methodologies have been developed to mitigate

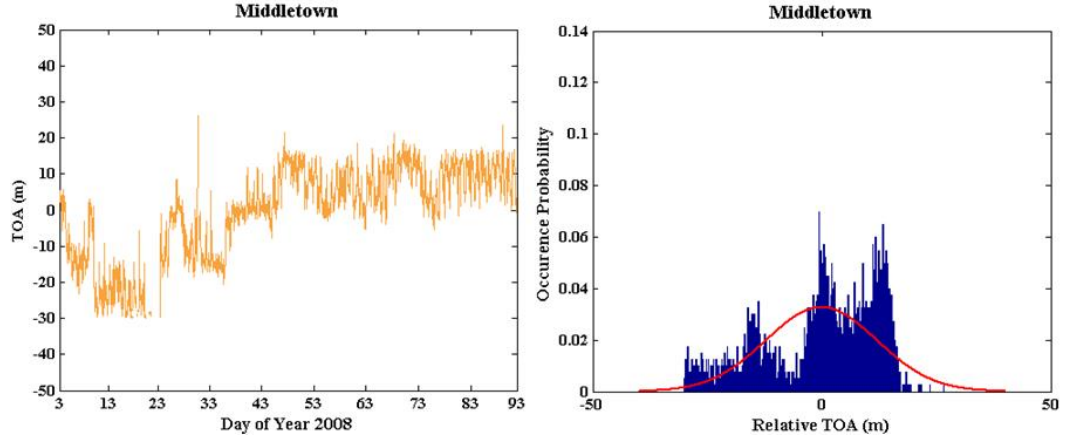


Figure 3.35: Middletown TOA measurements and its histogram

ASF. Two simple ideas are demonstrated in this dissertation: time difference and the “previous day is today’s correction.” Time difference (TD) refers to the difference in TOAs between secondary stations and the master station; thus, the master station is used as a reference to remove the ASF bias. The tradeoff using TD is that the total information entropy in a computed geotag is reduced due to the TOA information lost from the master station. In other words, using TD can achieve a high reliability or better geotag reproducibility but results in less information entropy or a shorter geotag. The second method is to use the previous day’s ASF measurements as today’s correction. This mitigation technique requires that either a user’s receiver constantly monitors Loran data or a reference station that is nearby broadcasts the previous day’s ASF as a correction via a secure data channel. The histograms of the corrected Middletown TOA using the two methods are plotted in the right plot of Figure 3.36.

The standard deviation of TD is 3.83 meters, whereas the “previous day is today’s correction” technique results in a standard deviation of 8.55 meters. Neither method can remove ASF completely. In the TD method, the different propagation paths of master and secondary stations result in a spatial decorrelation in the measured ASF. The previous day’s correction suffers from the temporal decorrelation of ASF because the previous day’s ASF differs from today’s ASF. If the ASF corrections from Loran reference stations can be updated frequently or broadcast in real time, for example, differential Loran corrections in LDC, the temporal decorrelation can be minimized.

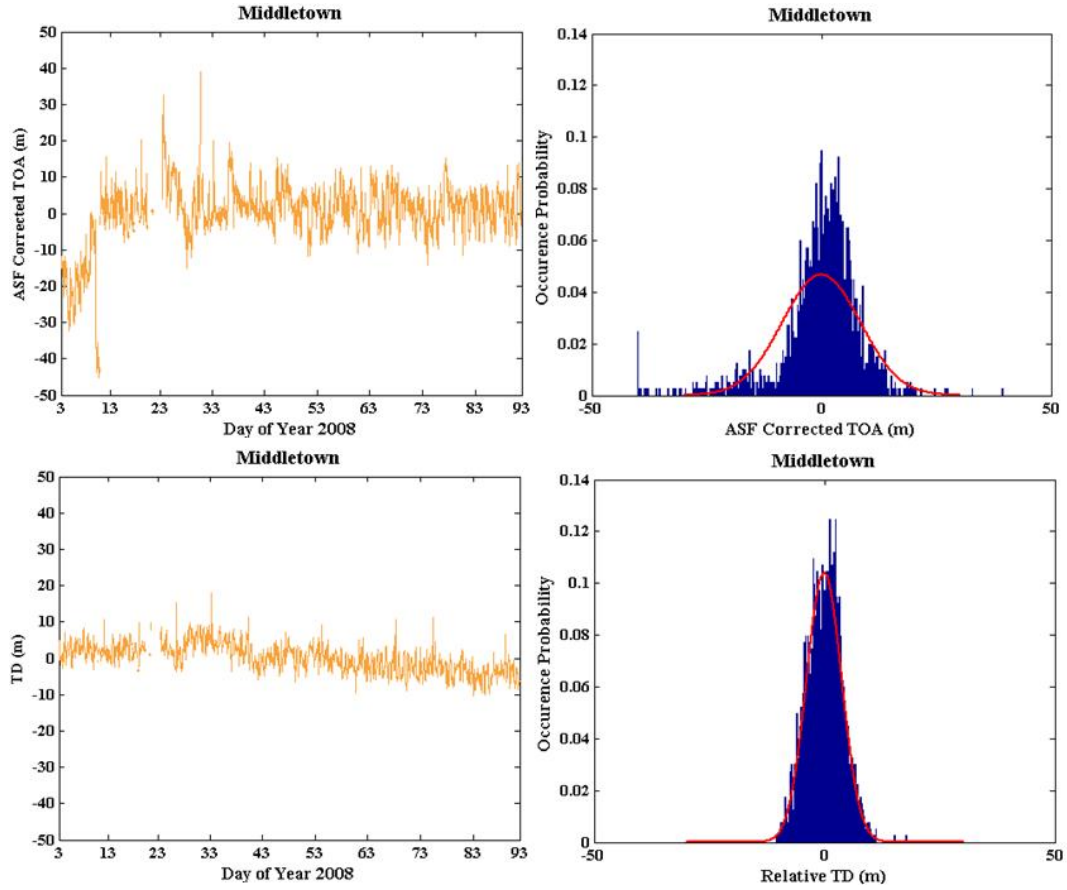


Figure 3.36: TOA correction: “Previous day is today’s correction” (top) and TD (bottom)

3.3.3 Continuity Evaluation Using Seasonal Data

The seasonal monitor data from GRI 9940 were used to evaluate the continuity of a geo-security system. The triple (TD, SS, ECD) data set was applied to compute a geotag. The first day of the 90-day data was taken as the calibration; the remaining days were considered to be the verification to estimate the FRR. The experimental FRR is the number of data samples in the 89 days, in which the geotags are matched with the one computed on Day 1, divided by total data samples. The quantization steps of various location-dependent parameters are calculated accordingly, based on the monitored standard deviation of the parameters. The FRR as a function of quantization steps is illustrated in Figure 3.37.

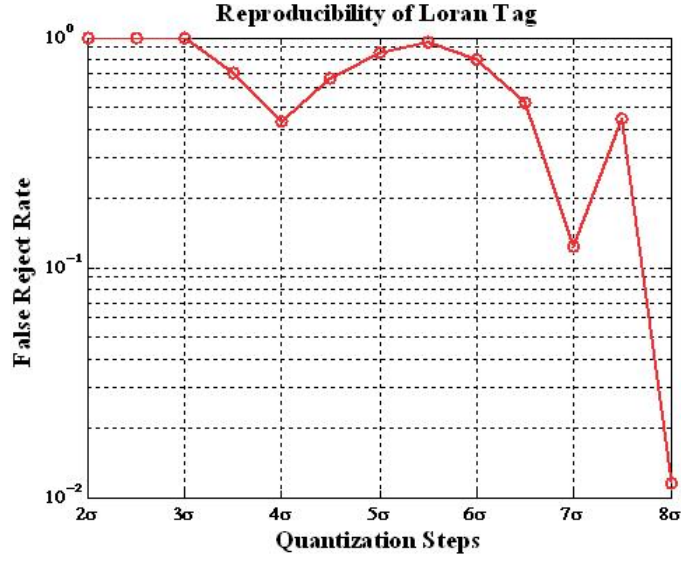


Figure 3.37: Reproducibility of a Loran geotag

Technically, multiple location-dependent parameters can achieve more entropy and higher geotag resolution, thus lowering false accept rate and raising the difficulty in predicting the desired geotag. However, one drawback of using multiple parameters is that either the FRR of the system is increased or geotag reproducibility is reduced. The system FRR can be estimated as $P_{FR} = \prod_{i=1}^n p_i$ assuming that the location-dependent parameters are independent from each other, where p_i is the error rate of one parameter, and n is the total number of parameters used to compute a geotag. Practically, location-dependent parameters have some correlations with one another in some environments. For instance, the signal strength is inversely proportional to the propagation distance, which is determined by TOA. In general, such a model is true when signals are not severely attenuated during propagation and the antenna is placed in an open sky area where there are no obstructions from the surroundings. The correlation is location, SNR, and parameter-dependent. As a result, the uncertainties complicate the modeling of the correlation between multiple parameters. Future research should examine the correlation of location-dependent parameters using data collected simultaneously at different locations. Sufficiently large data sets with temporally correlated components are required to model the correlation precisely.

To resolve the reliability problem using multiple parameters, an error-tolerant algorithm, fuzzy extractor, can be applied to improve geotag reproducibility. The construction and the performance validation of fuzzy extractors will be discussed in detail in Chapter 5.

3.4 Loran Information Measure

Although FRR and FAR quantify a parameter's repeatable accuracy and spatial decorrelation at one particular location, they cannot measure the location information density, which is also a key measure in the geo-security analysis. It is important to make sure that the location-dependent parameters can provide enough inter-location information or uncertainty in a computed geotag. This section uses the developed information theoretical approach to measure repeatable accuracy and spatial decorrelation of location-based information. "Location-based information" is referred to as the amount of information used to generate a geotag to identify one's location based on a set of measurements. The mathematical framework of the information measure is validated and evaluated using real Loran signals. Two data sets are collected to measure temporal and spatial entropy. Actual Loran data are helpful in evaluating the information theoretical approach because there are many practical concerns, such as local variations and receiver quality, which are difficult to predict and model mathematically.

3.4.1 Temporal Entropy

Equation (2.29) is applied to estimate the temporal entropy of Loran location-dependent parameters. Temporal entropy helps determine the parameter false reject rate but measures the inconsistency of location-dependent parameters in an information-theoretical approach.

Figure 3.38 illustrates how the temporal entropy varies with the FRR in three scenarios: TOA without ASF correction; TOA corrected using previous day's ASF correction; and, TD of Middletown. Each marker represents a different quantization

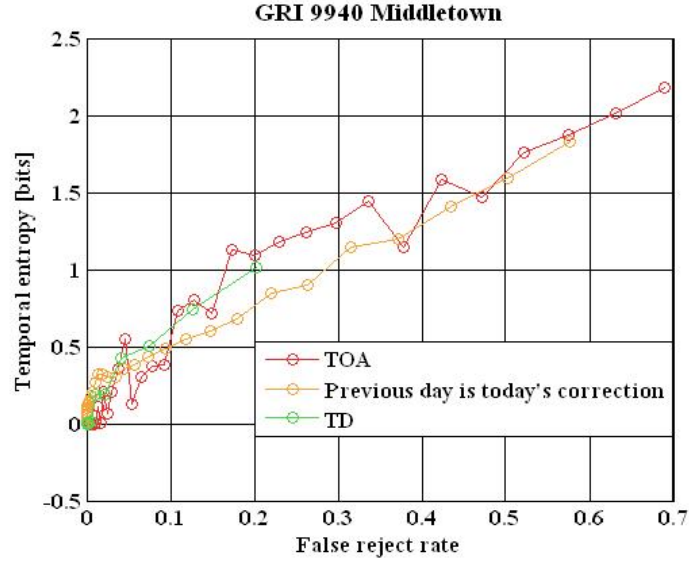


Figure 3.38: Temporal entropy as a function of FRR

step, which decreases from left to right, ranging from 100 to 10 meters. For the same quantization step, TD has smaller temporal entropy than the ASF corrected TOA.

3.4.2 Spatial Entropy

The spatial entropy was quantified using the same data sets from the locations shown in Figure 3.27. Considering the center point as the master location, the FAR decorrelation is examined as the test point is moving away from the center point.

Applying Equation (2.33) discussed in Section 2.4.2, it is easy to compute the spatial entropy of all the location-dependent parameters. Figure 3.39 illustrates the spatial entropy of the triple (TD, ECD, SNR) data set from four West Coast stations. Taking into account seasonal variations, the overbounded standard deviations and the quantization steps are applied to estimate the false accept rates. With the authenticated Loran signals and tamper-resistant device, the attackers cannot project or interpolate their locations to the target user's. All they can do is to rely on the measurement errors to eventually move their location estimate to the true location of the authorized user. The dark blue region has a spatial entropy 0 bit, which implies

that the attackers can easily achieve a correct geotag anywhere in this region. Attackers must make numerous attempts to map the received parameters into a correct geotag. Considering the authentication time discussed in Section 3.2.1, each attempt requires at least 38.4 seconds. For instance, if the spatial entropy of a location is 12 bits, attackers need to spend 43.69 hours to finish the trials of 2^{12} different attempts.

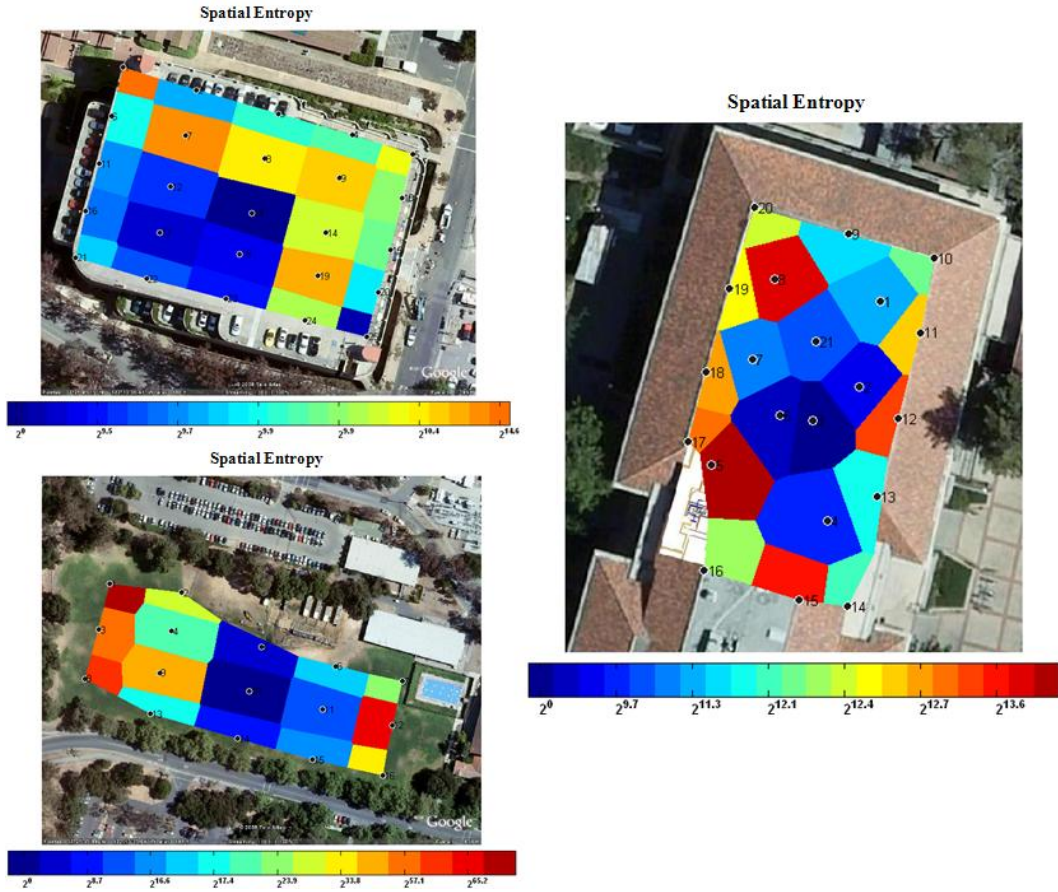


Figure 3.39: Spatial entropy: parking structure (upper left); soccer field (lower left); office building (right)

Parameters originating from high-SNR signals result in high spatial decorrelation due to small quantization steps. Based on the measurements, the validation of the uniqueness of the different location-dependent parameters is: TD has the highest spatial decorrelation; ECD has the least; SNR is very sensitive to environmental change due to the uncertainty of random noise.

3.4.3 Information Entropy to Bound Geotag Length

High information entropy can potentially result in a longer geotag as well as a higher security level for the system. This section uses the information-theoretical approach to provide an upper bound on the geotag over CONUS.

From Equation (2.34), it is easy to ascertain that high information content requires a large number of occurrences, N_i , but low temporal entropy, H_{T_i} , where N_i is the possible occurrences of a parameter computed from the range of each individual parameter and the selected quantization steps, and i represents each location-dependent parameter. Intuitively, if a parameter's temporal entropy is equal to its information entropy, the parameter cannot be used to compute a geotag. Nevertheless, as a parameter becomes more accurate, the location information content increases.

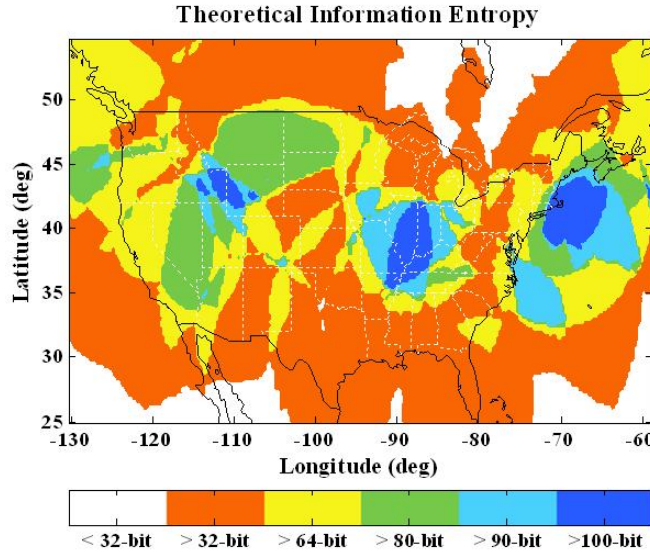


Figure 3.40: Geotag length upper bound

Overbounded quantization steps of location-dependent parameters can be obtained using standard deviation models. The signal strength model of 26 Loran stations, developed by Dr. Ben Peterson, was applied to estimate the entropy bound, assuming a constant noise floor for each GRI [3]. The quantization step, which is limited by the expected user performance (FRR) can be computed using the estimated standard deviation of location-dependent parameter and the desired FRR.

With overbounded step sizes, the temporal entropy is low or negligible; thus, the total information content depends only on the location information density.

Assume that the parameters are uniformly and independently distributed, the availability of information entropy over CONUS is illustrated in Figure 3.40. The FRR of 0.0001 for each location parameter is chosen to compute the step size. The information entropy varies spatially due to the different station coverage and the signal availability at each location. This analysis uses only the stations with the SNR higher than 3 dB, which is a lower limit for receivers to demodulate Loran messages properly and authenticate successfully, discussed previously in Section 3.2.1. For example, a user at Stanford University can achieve a 66-bit theoretical system entropy with an overall FRR of approximately 0.001.

3.5 Conclusion

This chapter has demonstrated and validated a geo-security system using Loran as a case study. A signal authentication scheme—TESLA—has been implemented and tested on Middletown, a Loran West Coast transmitter. The authentication performance depends on the received SNR, the data capacity, and the data modulation technique performed on LDC. Using the current U.S. proposal and access to 50% of the data capacity, it takes at least 38.4 seconds to authenticate a Loran signal source. A data collection trip was conducted from Stanford to Los Angeles to verify the authentication coverage; as a result, the signals at all eight test locations were authenticated successfully.

The analysis used the triple (TD, ECD, SNR) data set from four GRI 9940 stations on the West Coast to compute a geotag. Various data sets are used to evaluate the geo-security integrity, which is determined by the spatial decorrelation of location-dependent parameters, the resolution of a Loran geotag, as well as the robustness of the system to all of the possible attacks. Each location-dependent parameter offers different spatial decorrelation, and thus, different security strength. The experiments demonstrate that TD has the highest spatial decorrelation; a typical decorrelation distance of Middletown TD measurements is 18 meters in an open sky area. The

SNR is the most sensitive to the local environment which does not decorrelate much in open sky areas, such as a parking structure or a soccer field. The continuous spatial domain is converted into a discrete one by quantizing location-dependent parameters. The resolution of a derived geotag at a parking structure, a soccer field, and an office building is 17 meters, 40 meters, and 10 meters, respectively. With some degree of physical security, the location-based attacks cannot threaten a geo-security system. When an attacker is 20 meters away, the total attack time needed to break into the system is approximately 3.75 hours for the worst-case scenario, taking the 38.4 seconds to authenticate the signal source into account. As a result, a Loran geo-security system achieves high integrity. The authentication time can be reduced if a system with high data capacity is used for geo-security; thus, resulting in a reduction in the total attack time as well as the security strength. An artificial wait time can be injected to the system to retain the security level. The amount of wait time can be adjusted by users on tamper-resistant receivers.

The continuity of the proposed geo-security system was also studied using actual Loran data. Seasonal monitor data were used to examine the temporal variation of Loran location-dependent parameters and geo-security continuity. Continuity depends on both the transmitter operations and the receiver performance. The seasonal bias-ASF-must be corrected to minimize the TOA temporal variation. However, continuity risks have been observed due to both the transmitter operation and error sources on the receiver side. The solution to reducing continuity risks will be discussed and analyzed in Chapter 5.

Information entropy is another performance standard to evaluate a geo-security system since high information entropy results in a long geotag. Although Loran signals work well outdoors, they are significantly attenuated indoors due to walls, objects, and buildings. To complement Loran, Wi-Fi is chosen as our second case study to validate geo-security.

Chapter 4

System Design for Wi-Fi

Loran signals operate well outdoors; however, signal attenuation from obstructions in indoor environments degrades navigation and geo-security robustness. To complement Loran indoors, Wi-Fi is chosen as a second case study. Although Wi-Fi was initially designed for communications between electronic devices, the proliferation of Wi-Fi has spawned a growing interest in indoor location-based applications [23]. Figure 4.1 illustrates the integration of Loran and Wi-Fi geotag systems. Loran signals provide coarse resolution of spatial components. Time messages in the Loran data channel provide an additional dimension in the geotag generation; authentication messages ensure the integrity of Loran signals and protect against spoofing. On the other hand, the resolution of Wi-Fi geotags is finer than that of Loran geotags. As a result, Wi-Fi improves the spatial discrimination and resolution of a derived geotag as well as the information entropy.

4.1 Wi-Fi Signal Characteristics

The growing deployment of Wi-Fi devices by individuals and organizations in homes, offices, and campuses presents an opportunity for Wi-Fi indoor positioning. Most mobile devices, such as laptops, PDAs, and cellular phones, are equipped with Wi-Fi devices [26]. The infrastructure can be used to provide indoor location-based applications without deploying additional equipment. One drawback of Wi-Fi positioning

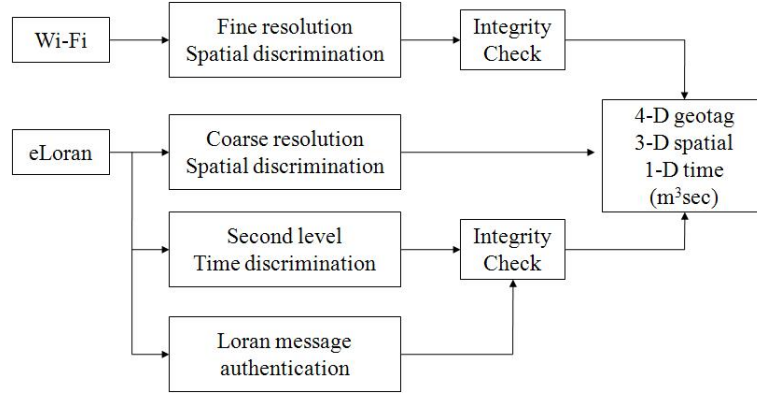


Figure 4.1: Integration of Loran and Wi-Fi for geo-security

systems is limited coverage due to the transmitting range of access points (APs). Therefore, Loran compensates for the coverage of Wi-Fi, while Wi-Fi assists in the spatial decorrelation of a Loran geotag.

Many Wi-Fi positioning systems use the received signal strength (RSS) and/or the medium access control (MAC) address from nearby access points to derive symbolic locations. A symbolic location refers to the proximity of known objects or abstract ideas of location [7, 23] rather than physical coordinates such as latitude and longitude. For example, a symbolic location representation could be “Joe Laas is in room 450 on the fourth floor of the Durand Building.” Both the MAC address and the RSS are location-dependent parameters, and can be used to generate a Wi-Fi geotag.

To receive the Wi-Fi location-dependent parameters and learn the signal characteristics in various environments, a portable data collection setup is built. The data collection equipment consists of a Wi-Fi-enabled laptop and a Garmin GPS receiver. The software (WirelessMon) periodically scans the environment and records the tracked AP MAC addresses and RSSs to a log file. Users can adjust software settings using the interface illustrated in Figure 4.2. The software appends the recorded latitudes and longitudes to the log file when the GPS receiver is attached to the Wi-Fi device.

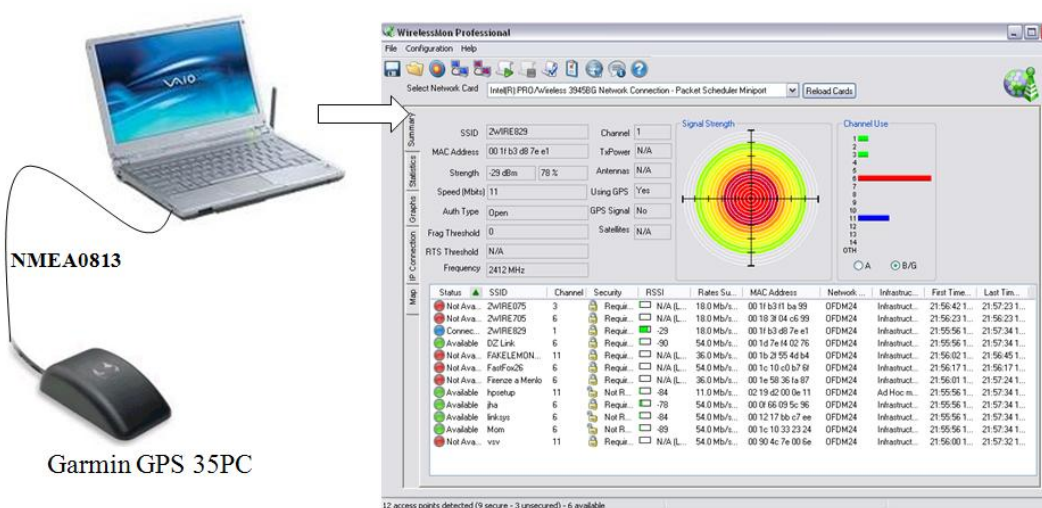


Figure 4.2: Wi-Fi data collection setup

AP Density in a Downtown Area

The first experiment examines the spatial decorrelation and coverage of Wi-Fi APs. Downtown Menlo Park, pictured in the top of Figure 4.3, was chosen for this test, which required a user to drive around the neighborhood with a Wi-Fi-enabled laptop and the Garmin GPS receiver. The driving paths are plotted in green and superimposed on a Google map. The results indicate that the AP density in the area is approximately 1155/km². To examine the spatial decorrelation of Wi-Fi APs, the center point, indicated as a red marker, was selected as the master location. Separations between other points and the center point were calculated using the recorded latitudes and longitudes. The percentage of APs that are shared with the reference location represents the AP spatial decorrelation. When a test point is 200 m away from the center point, it does not share any tracked APs with the center point. Even with 10 m separation, the percentage of the shared APs is approximately 86%. At most of the test points, the device can track more than four APs, as illustrated in the histogram on the bottom right of Figure 4.3.

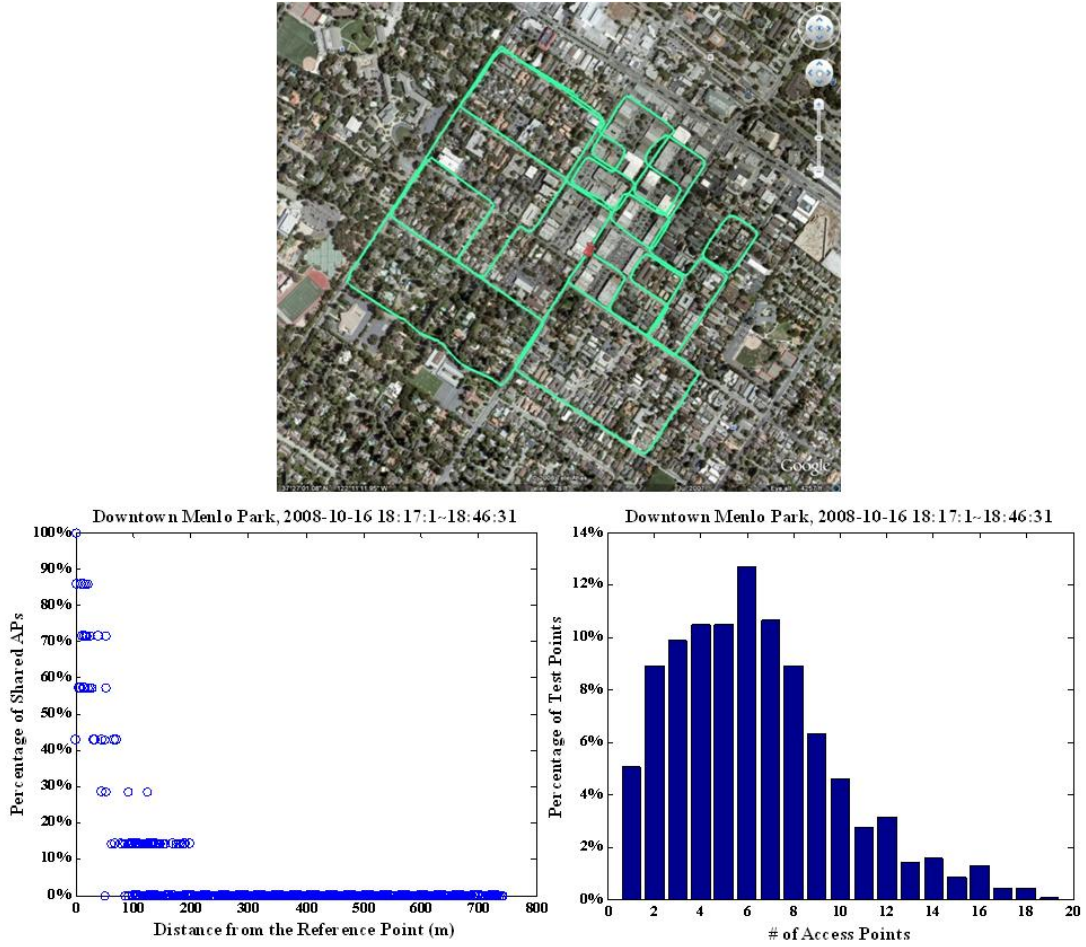


Figure 4.3: Downtown Menlo Park (top); Spatial decorrelation (bottom left); AP spatial distribution (bottom right)

Residential Area

This section studies the Wi-Fi signal properties using the measurements collected in a residential area for eight hours as depicted in Figure 4.4. The AP with the strongest RSS represents the connected node. The RSS measurements have temporal variations in the range of 10 dBm or less. Generally speaking, stronger APs have fewer temporal variations in RSS. Both thermal noise and multipath can contribute to the temporal variations of the measured RSS. The multipath fading effect is the result of destructive or constructive combination of multiple signals at a receiver and

causes the signals to fluctuate around a mean value. Thus, multipath is a common error source in indoor environments due to signal refraction, reflection, and diffraction from the environment.

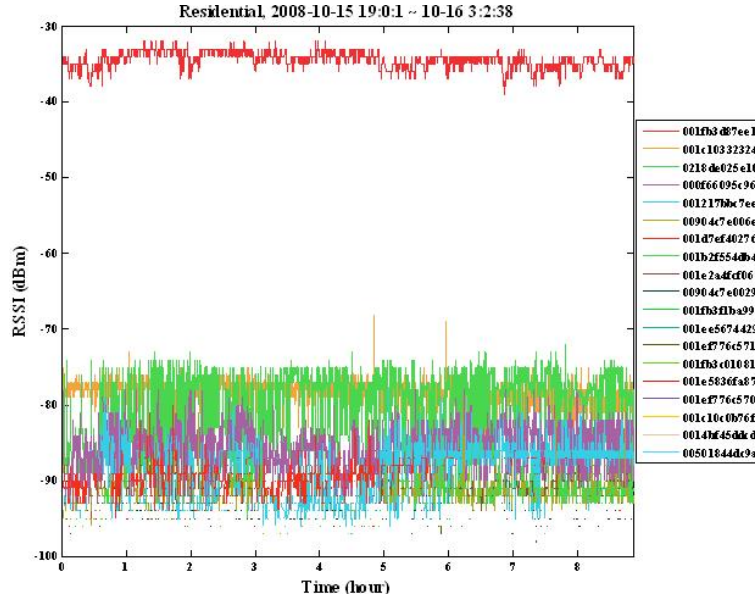


Figure 4.4: Wi-Fi RSS measurements in a residential area

Signal strength is a common metric to determine the propagation distance from a radio source in many RF-based systems. A propagation model is needed to convert the signal strength to the propagation distance. To characterize the model, RSS readings were collected for 3 minutes at varying distances from an AP. Figure 4.5 depicts the Wi-Fi RSS as a function of distance between the observed AP and the receiver. Each dot represents the mean of the collected RSS measurements at a particular test point. The correlation between RSS and the distance is visualized. In practice, it is difficult to design a mathematical propagation model that is suitable for all environments, especially indoors because signal propagations are not linear. The signal attenuation originates from the path propagation, reflection, diffraction, diffusion, and transmission through various materials [13]. Moving objects such as people can cause not only attenuation but also fluctuation. The sum of all the components is taken to obtain the RSS; as a result, RSS varies temporally. Therefore, the conversion from RSS to the propagation distance is valid only when the signal strength

attenuation is predictable, and there is no extra attenuation from the composites of walls, building structures, moving objects, and multipath effects, etc. For instance, the attenuation factor differs for brick walls, wood, and glass. A structure composed of a variety of materials or composites complicates the modeling of the attenuation factor.

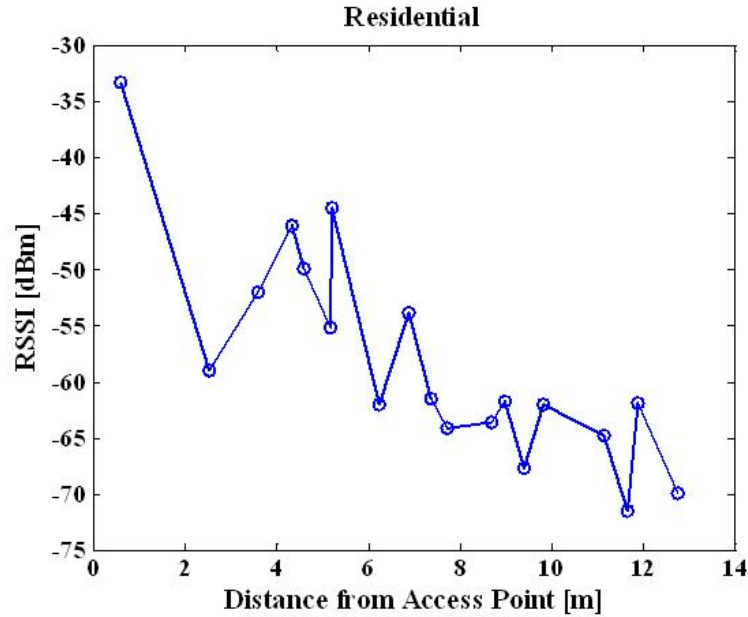


Figure 4.5: RSS as a function of distance between AP and receiver

Office Building

A second data set was collected for four hours in an office building. This is the same office building in which the Loran experiments were conducted. The RSS measurements fluctuate more in this environment than in the residential area, since there are more sources of attenuation and signal blockages in the office due to moving people. The human body is composed of a large percentage of water, which has a resonance frequency at 2.4 GHz and greatly attenuates the Wi-Fi signals [23].

The visualization of the quantized cells created from the computed geotags is illustrated in Figure 4.6. Only MAC addresses were used for the geotag generation,

that is, $T = MAC_1 || MAC_2 || \dots || MAC_m$, where m is the total number of APs used. Similar to the Loran geotags shown in Figure 3.27, the two-dimensional cells were created using Voronoi diagrams, and colors were mapped into the cells based on the computed geotags accordingly. The geotags computed from four APs have an average cell diameter of 5.7 m; the geotags computed from eight APs result in a slightly larger average cell diameter of 6.7 m.

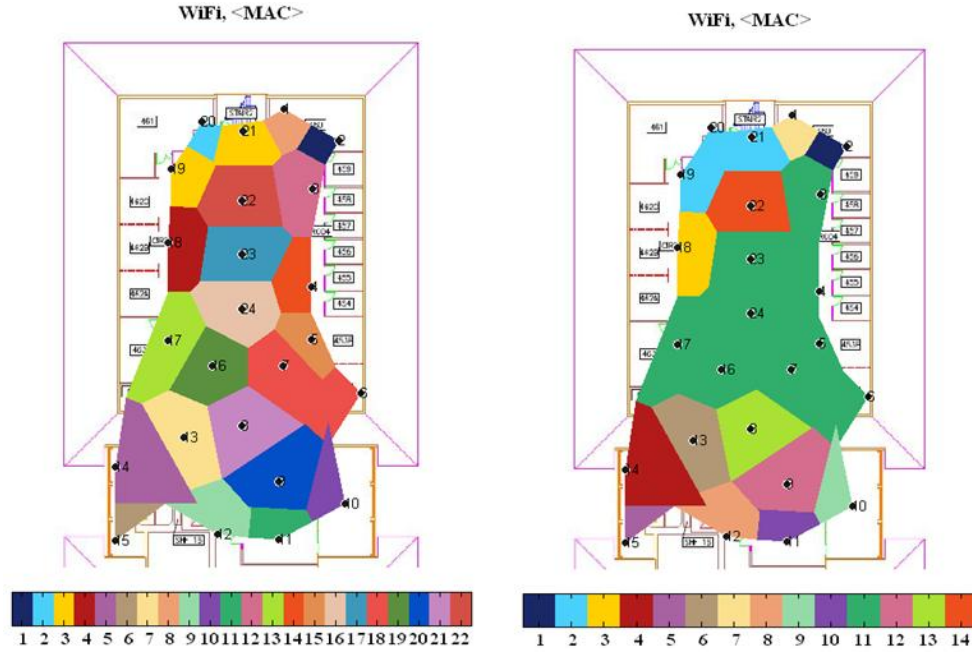


Figure 4.6: Geotag visualization in an office building: eight APs (left); four APs (right)

4.2 Continuity Analysis

The availability or response rate of an AP is defined as the percentage of time that a receiver is able to track the AP [42]. A set of Wi-Fi scans from an AP are collected, and the availability can be computed based on the fraction of times that the AP is observed and the total number of scans. The right plot in Figure 4.7 indicates a correlation between the AP availability and the RSS. As the receiver is close to an AP, it is expected that the availability will be high; as the receiver moves away from

the AP, there is more attenuation and the availability is lower. However, the accuracy of the availability measure is limited by the total number of scans and the quality of the Wi-Fi receiver. Thus, more scans will provide better estimates. The quality of a Wi-Fi receiver also plays an important role in capturing, conditioning, and processing the received signals.

Residential Area

The left plot in Figure 4.7 illustrates the availability of all of the APs tracked during the eight hours. The first four strongest APs have relatively high availability. If more than four APs are used to compute a geotag, there is no guarantee that the geotag will be reproducible at a later time. In other words, the false reject rate will be high.

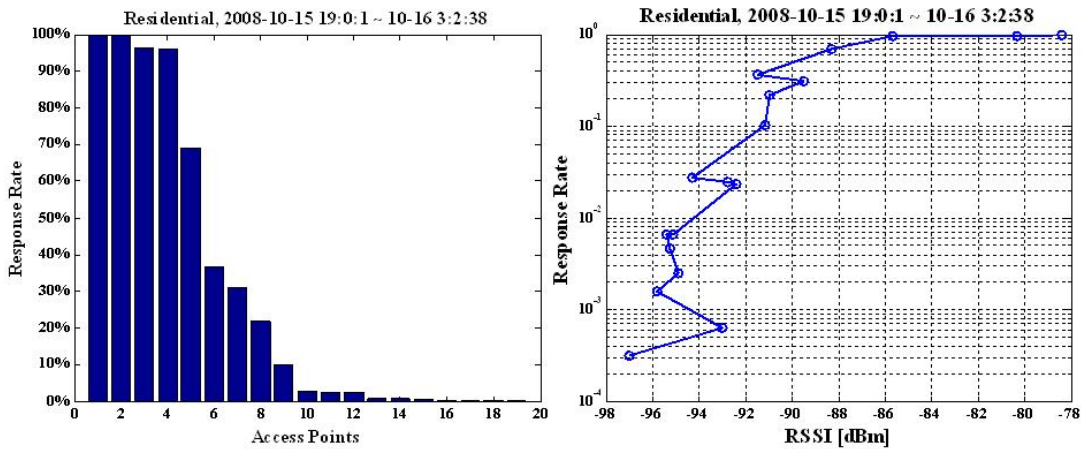


Figure 4.7: Residential: availability histogram (left); availability as a function of RSS (right). Low RSS signals are easier to lose track.

The FRR and geotag resolution can be traded off against each other by varying the number of APs used to generate a geotag, as illustrated in Figure 4.8. A larger number of APs provides high spatial decorrelation or discriminations, resulting in a small quantized space. However, increasing the number of APs also increases the likelihood of low-availability APs and lowers the reliability of the system. For instance, with eight APs, a cell diameter of 3.5 m is achieved, but the derived geotag has an extremely poor reproducibility, that is, the FRR almost goes to 1. In this scenario,

the number of APs for geotag generation should not exceed four. With four APs, the optimal performance is achieved: the FRR is reasonably low, and the cell diameter or the resolution of the geotags is relatively small.

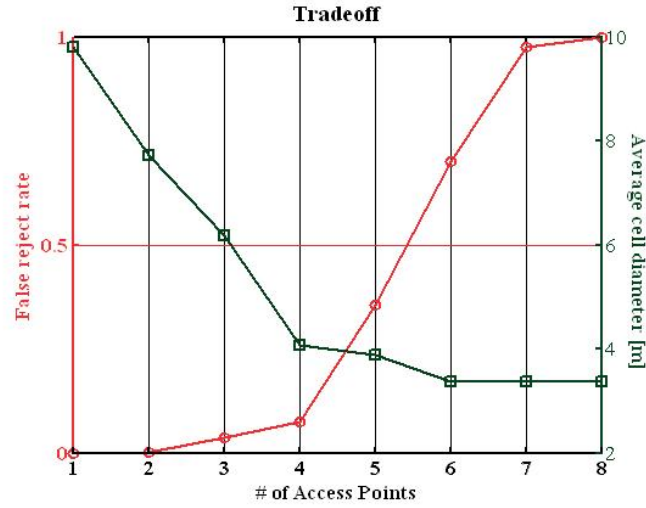


Figure 4.8: Residential: Tradeoff between FRR and geotag resolution

Office Building

Applying the same office building data, the geotags are computed from four APs, and the average cell diameter is approximately 7 m. The availability of observed APs is illustrated on the left of Figure 4.9. Although the total number of APs tracked in the office is more than that of the residential area, the availability of the tracked APs is poor. The same tradeoff analysis is conducted to examine the optimal number of APs required to generate a geotag. The FRR is relatively high compared with the residential FRR. To achieve a reproducible geotag, only the AP with the strongest RSS can be used. Even with two APs, the geotag reproducibility is slightly high, that is, FRR is 0.15. The average cell diameter is reduced from 10.3 m to 6 m as the number of APs increases from one to eight.

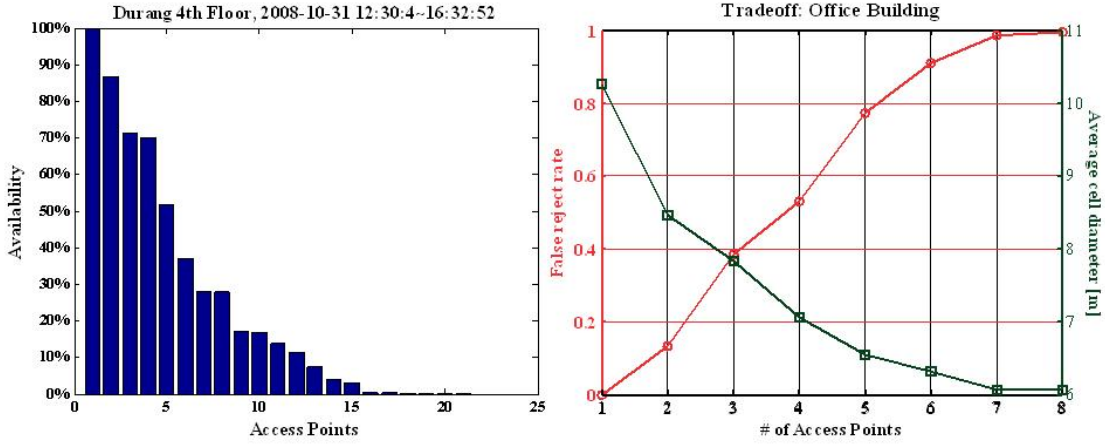


Figure 4.9: Performance analysis in the office building

4.3 Conclusion

The integration of Loran and Wi-Fi produces better performance (low FAR and FRR) in the geotags since more parameters increase high spatial discriminations. In addition, increased parameter diversity, greater RF signal variety, and a larger number of transmitters improve the information entropy in the derived geotags and the position estimation robustness. It is not necessary to synchronize the different systems at either the transmitter or receiver ends. The performance of Loran and Wi-Fi integration is illustrated in Figure 4.10. The geotags are generated from the MAC address and RSS of four Wi-Fi APs. The Loran location parameters are TD, SS, and ECD from four West Coast stations used for the geotag generation. The resulting quantized cells have a minimum size of 2.7 m and average size of 6 m. The average cell diameter is reduced by 32% with the addition of Wi-Fi signals.

The properties of Loran signals can improve the design of location-based security services and geotag-approached positioning, whereas Wi-Fi signals complement Loran with more spatial variations, greater signal variation and information entropy in indoor environments. Therefore, the integration of two signals results in higher integrity in geo-security.

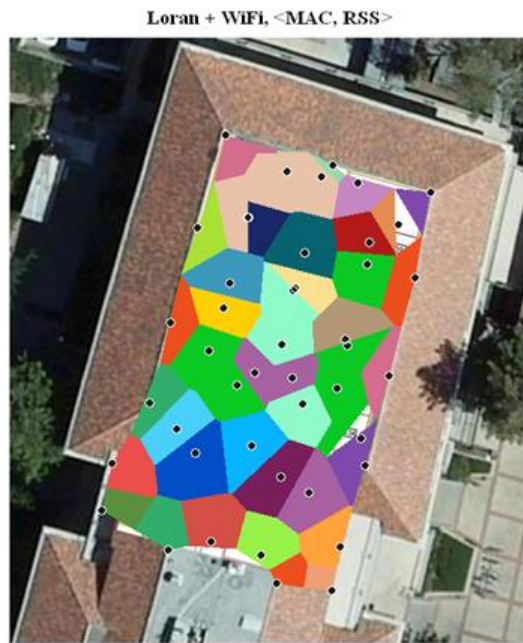


Figure 4.10: Loran and Wi-Fi integration

Chapter 5

Fuzzy Extractors to Reduce Continuity Risk

Reproducibility and repeatable accuracy are desirable qualities in geo-security systems. They permit one to provide location-dependent parameters, or the derived geotag at calibration—and still retain the same valid parameters at a later time for verification. However, the continuity risks, addressed in Sections 3.3 and 4.2, increase the likelihood of failure to validate authentic users. This chapter focuses on the solutions to continuity risks originating from the transmitter operation and measurement errors.

5.1 Continuity Risks

The signal characteristics should be consistent enough that when a user is ready to verify, measurements at the same location will yield the same previously-generated tag. Temporal variation reflects the instability or some degree of scatter within a particular parameter at a given location and causes valid geotags to be mismatched with the calibrated geotags in the database. Therefore, error-tolerant algorithms must be applied to reliably extract location information from noisy RF signal inputs. A fuzzy extractor (FE) is a form of error-tolerant algorithm that reproduces the desired secret information. The extraction in a fuzzy extractor is error-tolerant in the sense

that the derived geotag remains the same even if the input changes slightly.

5.1.1 Error Model

To achieve optimal construction of fuzzy extractors, various types of errors presented in location data were studied [36]. Several possible error sources in RF signals are random noise, atmospheric noise, seasonal bias, and multipath. Thermal noise, considered as white Gaussian, cannot be eliminated and is always present in all electronic devices and transmission media. Atmospheric noise, caused by lightning, is non-Gaussian and dominant in low-frequency signals, and can be impulsive if the lightning is local. Both random and atmospheric noise can be affected by the transmitter radiating power, the propagation path, the propagation distance, the quality of the receiver, and the local noise floor.

An important error source in Loran signals is a seasonal bias, named Additional Secondary Factor (ASF), as discussed in Section 3.3. This error introduces large seasonal variations in the measured TOA, as shown on the top left of Figure 3.35. In this dissertation, the Time Difference (TD) method is applied to mitigate partial ASF temporal variations, although it suffers from the spatial decorrelation introduced by the different propagation paths of master and secondary stations [39].

In addition, the quantization error, which is the difference between the value of a continuous parameter and its quantized value, can cause the system to fail to reproduce a correct geotag. The quantization error is usually correlated with the thermal noise, the atmospheric noise, and the seasonal biases discussed above. As a result, it is not guaranteed that the measurements are always in the middle of the quantization grid. The worst-case scenario is that the measurements lie on the boundary of the grid, as depicted in Figure 5.1. The graph plots the TD measurements from Middletown with zero mean. The red dashed lines represent the quantization grid boundaries. Even though the quantization step is selected to overbound signal variations caused by random noise and seasonal biases, the quantization error increases the likelihood of failure to reproduce a correct geotag.

The last type of error originates from the operations of the RF system. Loran

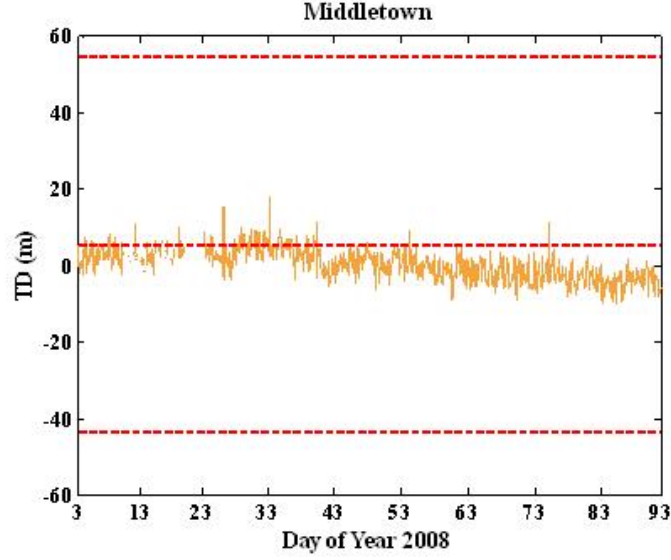


Figure 5.1: 90-day Middletown TD measurements with quantization grids, $\Delta = 50$ m

stations might be offline due to maintenance or other implementation issues. Wi-Fi access points (APs) are moved around or turned on/off by their users. The Wi-Fi AP availability or response rate illustrated in Figure 4.7 shows that only the first AP can achieve a 100% response rate; thus, a geotag will not be reproducible if more than two APs are used to derive a geotag.

5.1.2 Distance Measures

The geotag reproducibility under natural variations of the RF signal is relative to the underlying metric in the space of the location data. Thus, it is important to analyze the error patterns and determine the proper distance metrics accordingly since the construction of fuzzy extractors depends on the distance metrics of the inputs.

The same variables introduced in Chapter 2 apply here. Let x be the location parameter vector at the calibration step, and q_x be as defined in Equation (2.2). $q_i = \mathcal{E}(x_i) = k; x_i \in S_k = [k\Delta_i, (k+1)\Delta_i)$, $k = 1, \dots, N$ represent the discrete parameter vector after the quantization. The pair (x', q'_x) represents the parameter vectors at the verification step. Δ is the quantization step vector. All of the vectors are n -dimensional, where n is the number of parameters used to compute a geotag.

Quantized parameters, q_x and q'_x , are integers over Z , but they are not necessarily positive. For instance, TD will be negative if the distance between the secondary station and a user is shorter than the distance between the master station and the user.

The most common metric for the location parameter vector x is Euclidean distance. A Euclidean metric fuzzy extractor is designed to tolerate the random noises, seasonal biases and quantization errors.

The distance measure for the last error type (a missing parameter or offline transmitter) is considered to be a Hamming metric. For example, even if one or more transmitters fail to broadcast during user verification, it is desired that the derived geotag is error tolerant and can still successfully validate the user location, that is, a subset of the location-dependent parameters can reproduce the geotag computed at the calibration step. Hamming distance measures the number of different elements in a quantized location parameter vector at the calibration and the verification, q_x and q'_x , where $|q'_x| \leq |q_x|$.

5.2 Fuzzy Extractor

In geo-security, fuzziness refers to the instability of the location-dependent parameters induced by the various error sources discussed in Subsection 5.1.1. This section designs various types of fuzzy extractors, which play an important role to reproduce the same previously generated geotags even if $x' \neq x$ and $q'_x \neq q_x$. As a result, fuzzy extractors improve the FRR and reduce the system continuity risks. This section also analyzes the tradeoff between FAR and FRR after implementing fuzzy extractors.

The first approach of a fuzzy extractor or an error-tolerant cryptographic algorithm, called a *fuzzy commitment scheme*, is proposed for biometrics by Juels and Wattenberg [21, 22]. The scheme uses an error-correcting code to handle the Hamming distance. Other approaches for Hamming distance, set difference, and edit distance are introduced by Dodis et al. in [18]. They also introduce a different error-tolerant algorithm, called secure sketch.

5.2.1 Definitions

This section follows the definition of fuzzy extractors in [18]. A fuzzy extractor operates in two steps, illustrated in Figure 5.2. During the calibration step, an algorithm Gen is run on an input, $x \in M$, to generate a public value, P , and a geotag, T , where M is a metric space of x . x is the vector of expected signal characteristics based on a calibration of the location to be secured. The public value, P , is stored for future use. The computation of P varies with the constructions of fuzzy extractors. An algorithm, Rep , is used to reproduce the geotag, T , using P from noisy location vector x' . Fuzzy extractors have been proved to be information-theoretically secure; thus, they can be used for security applications without introducing additional assumptions [18]. A secure sketch also consists of two steps. A procedure, SS , produces s , called a *sketch*, using an input x . Then, given s and x' close to x , a procedure, Rec , can recover the input x . The sketch s should not reveal much information about x . Unlike fuzzy extractors, a secure sketch recovers the original input x from noise, whereas a fuzzy extractor reproduces geotag, T , from a noisy input.

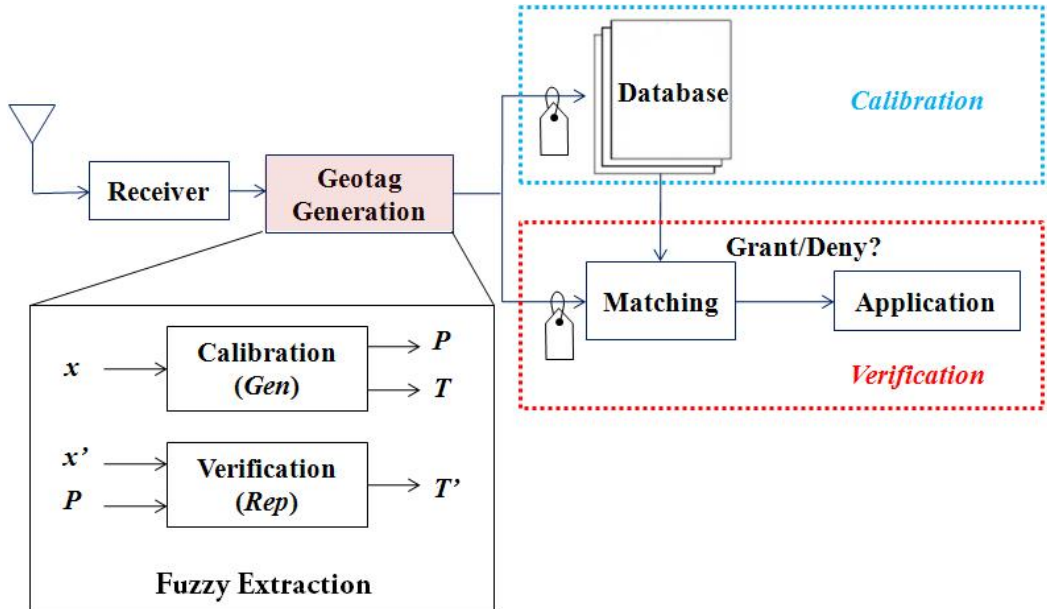


Figure 5.2: Fuzzy extractor construction for geo-security

Definition 1. A fuzzy extractor is a tuple (M, t, Gen, Rep) , where M is the

metric space with a distance function, dis , Gen is a generate procedure and Rep is a reproduce procedure, which has the following properties: If $Gen(x)$ outputs (T, P) , then $Rep(x', P) = T$, whenever $dis(x, x') \leq t$. If $dis(x, x') > t$, then there is no guarantee T will be output.

Definition 2. A secure sketch is a tuple (M, t, SS, Rec) , where M is the metric space with a distance function, dis , SS is a sketch-generating procedure, and Rec is a recover procedure, which has the following properties: $Rec(x', SS(x)) = x$, if $dis(x, x') \leq t$. The sketch s is to be made public. The scheme is m -secure, and the entropy loss of s is at most m . $H(x) - H(x|s) \leq m$. H denotes the entropy of a random variable.

In this dissertation, four fuzzy extractors are proposed based on the different distance metrics for temporally inconsistent location-dependent parameters: Euclidean metric, Reed-Solomon-based Hamming metric, Secret sharing-based Hamming metric, and secure sketch-based Hamming metric fuzzy extractor. The fourth approach introduced is the secure sketch that Chang and Li [11, 10, 12] proposed for small set difference. Although their construction was initially designed for biometric data, it can be adapted for location data with modifications. The modified secure sketch works the same as the Hamming metric fuzzy extractor for location data. These constructions differ in the amount of public information, decoding efficiency, and the degree of ease in practical implementation. The appropriate fuzzy extractor construction should be applied depending upon the RF signals to implement geo-security, computation power, and the users' decision on the FRR-FAR tradeoff.

5.2.2 Euclidean Metric Fuzzy Extractor

The Euclidean fuzzy extractor generates the closest geotag from the noisy data. Let the location vector, x , be n -dimensional in metric space M . Consider the distance measure for the location-dependent parameters to be L_∞ norm, which is well adapted to the handling of distance bounds. The measure can be normalized using Δ ; the distance is defined as:

$$dis(x, x') = \left(\max_i \frac{|x_i - x'_i|}{\Delta_i} \right)_{i=1}^n. \quad (5.1)$$

The basic idea of this fuzzy extractor is to adjust the offsets between the continuous measurements and their discrete values following the quantization process. The construction of the fuzzy extractor is illustrated as follows:

$$Gen(x) = \left(\begin{array}{l} T = hash(q_x) \\ P = \left(x_i - \Delta_i \left\lfloor \frac{x_i}{\Delta_i} \right\rfloor \right)_{i=1}^n \end{array} \right), \quad (5.2)$$

$$Rep(q'_x, P) = \left(\begin{array}{l} q'_x = \left\lfloor \frac{x_i - P_i + \frac{\Delta_i}{2}}{\Delta_i} \right\rfloor_{i=1}^n \\ T' = hash(q'_x) \end{array} \right). \quad (5.3)$$

If $dis(x, x') < \frac{1}{2}$, then the quantized location vector q'_x can be reproduced, that is, $T' = T$. This claim defines the reproducibility of a geotag. The quantization step Δ is a design parameter. The bigger the step, the more errors can be tolerated using this fuzzy extractor.

A working example of the Euclidean metric fuzzy extractor using Loran TOA measurements is illustrated in Figure 5.3. At the calibration step, the TOA measurements from the first day, plotted in orange, are quantized; the public information, P , is derived from the TOA mean value and the quantization offset. P is then saved for the verification use. The blue dotted lines indicate the quantization grid boundaries. At the verification step, which is 30 days after, the mean of the TOA measurements shifts down to the bottom quantization grid. After applying the public information, P , the offset introduced by the errors and bias is minimized; thus, the quantized value of TOA is reproduced.

Shannon entropy is used to measure the entropy loss of fuzzy extractors mathematically. The entropy loss or the mutual information between the conditional $H(x|P)$ and unconditional $H(x)$ entropies is estimated. If the mutual information is zero, they are statistically independent. Given $x = q_x + P$, let $x' = q_x + P - \delta$, where δ is the Euclidean difference between x and x' due to the noise and biases. The objective is to determine an upper bound on $H(x|P)$. By using the definition of conditional

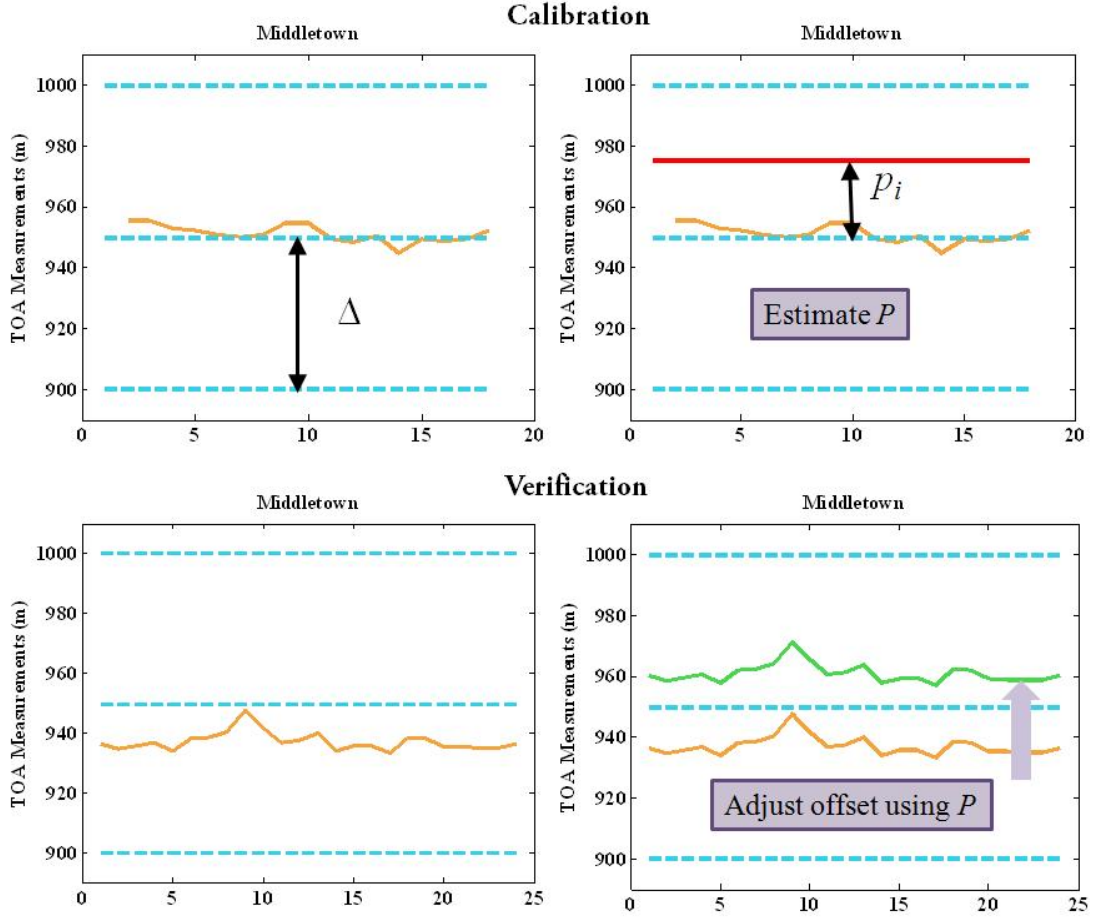


Figure 5.3: A demonstration of Euclidean metric fuzzy extractor using Loran measurements: calibration data (top); verification data, 30 days after (bottom).

entropy [14],

$$H(x|P) = H(x) - H(\delta) \quad (5.4)$$

Thus, the entropy loss of the public value, P , is $H(\delta)$, which depends on the probability distribution of x and the quantization step Δ . For n number of different location parameters, the total information leakage is

$$H(\delta) \leq \sum_{i=1}^n \log(\Delta_i). \quad (5.5)$$

The above equation assumes that the parameters are uniformly and independently distributed and provides an upper bound on the entropy loss. In practice, the entropy loss is small in comparison with $H(x)$. The measured entropy in a geotag also quantifies the amount of uncertainty from an attacker's point of view. The entropy in a geotag computed from the quantized parameters equals $H(q_x|P)$. From the definition of q_x , it is independent of P ; thus, P does not leak any information on q_x . Intuitively, it makes sense that knowing the offsets between x and $\Delta_x q_x$, one cannot predict or guess the user's quantization level exactly without further information.

5.2.3 Hamming Metric Fuzzy Extractors

Reed-Solomon Based

The approach achieves robustness against noise and biases by utilizing error-correcting codes to recover changes measured by the Hamming distance. The Hamming distance, defined in Equation (5.6), measures the number of different elements between two strings or vectors. In addition, this fuzzy extractor deals with the problem caused by offline transmitters. A geotag can be reproduced even if there are missing parameters. The fuzzy extractor described in the last subsection does not have this ability to handle missing data.

$$dis(x, x') = \sum_{i=1}^n x_i \oplus x'_i \quad (5.6)$$

A Reed-Solomon (RS) error-correcting code is used to construct a fuzzy extractor to recover the changes of the quantized location parameters. Reed-Solomon coding is a well-known forward error correction coding method that is designed for burst errors [15]. The key idea of the construction is to first create a polynomial by encoding the secrets, which is the geotag in geo-security systems. The next step is to project the quantized location-dependent parameters on the polynomial and randomly create chaff points to hide the polynomial. Finally, the secrets can be recovered from the chaff points with adequate location parameters. The detailed construction is described as follows:

Calibration. Given $q_x = \{q_1, \dots, q_n\}$,

1. A secret message is computed from a random generator.
2. Unlike the quantization-based geotag generation discussed in Section 2.1 where the geotag is created from the location measurements, the fuzzy extractor-based geotag is computed from the secret message using a random generator.
3. The geotag, T , is encoded to a vector, c , using Reed-Solomon code. The vector, c , has a size of n . The RS encoder (n, k) is chosen based on the design criteria, that is, the total number of errors, t , that can be corrected is determined by $\frac{n-k}{2}$.
4. Construct the mapping matrix or the public information, P . P has the size $N \times n$, where N is the number of quantization levels of location-dependent parameters and is determined by the chosen quantization steps. For each column of P , locate the element of vector c based on each quantized location-dependent parameter. For instance, if $q_i = 20$, then $P(20, i) = c_i$. Figure 5.4 demonstrates the formation of a mapping matrix, P . Populate the remainder of the matrix using random numbers. This mapping matrix is then saved for future use.

$$Gen(q_x, m) = \begin{pmatrix} T = rand \\ c = RS\ encode(T) \\ P = Mapping(c, q_x) \end{pmatrix} \quad (5.7)$$

Verification. Given that q'_x is a location parameter vector that has t or less than t elements that differ from q_x .

1. Obtain the mapping matrix, P , generated previously.
2. Derive a vector, c' , using P and q'_x . If q'_x and q_x are identical, c' contains the same elements as c . If attackers know little or no information about the location-dependent parameters, q_x , then it is difficult to guess a vector c' that satisfies $dis(c, c') \leq t$ due to the large search space of the mapping matrix. It is equivalent to a brute-force attack.

q_3			c_3			
q_2		c_2				
q_1	c_1					
	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
	1	2	3	4	...	n

Figure 5.4: Mapping matrix for an RS-based fuzzy extractor

3. Apply the Reed-Solomon decoder to compute T from c' . If $\text{dis}(c, c') \leq t$, the secret message can be recovered correctly; otherwise, the output will not be the same as T .

$$\text{Rep}(q'_x, P) = \begin{pmatrix} c' = \text{Mapping}^{-1}(P, q'_x) \\ T' = \text{RS decode}(c') \end{pmatrix} \quad (5.8)$$

This approach makes use of the Reed-Solomon code property to tolerate t errors in the quantized location-dependent parameters. It is not fault-detective since users would not be able to ascertain whether the errors in received location parameters can be tolerated or not until the computation of a geotag is accomplished. The entropy loss of this construction is $t \log N$. Such a loss results in an effective geotag length, $(n - t) \log N$. Thus, Hamming metric fuzzy extractors improve geotag reproducibility at the expense of entropy loss.

Secret Sharing Based

Another construction of a Hamming metric fuzzy extractor is based on the idea of secret sharing. This scheme is a method of sharing secret, S , among a set of n participants. For any subset of k ($k \leq n$) participants, the secret S can be reconstructed. On the other hand, a subset of less than k participants will fail to reconstruct S .

The distance metric in this construction is also Hamming. The input to the fuzzy extractor is the quantized location vector, q_x . The first construction step is to

create a polynomial, $f(x)$, such that $f(i) = q_i, \forall i = 1, 2, \dots, n$. The generation and reproduction procedures are as follows:

$$Gen(x) = \begin{pmatrix} f(i) = T + a_1x + a_2x^2 + \dots + a_kx^k \\ a_1, a_2, \dots, a_k \text{ are random numbers} \\ P = \langle i_1, \dots, i_k \rangle, \text{ s.t. } f(j_j) = q_j \end{pmatrix}, \quad (5.9)$$

$$Rep(x', P) = \begin{pmatrix} \text{Reconstruct } f(i) \text{ using } P \text{ and } q'_x \\ T' = \langle f(0) \rangle \end{pmatrix}. \quad (5.10)$$

If $dis(q_x, q'_x) \leq n - k$, the polynomial, $f(x)$, can be reconstructed with the assistance of P ; thus, the desired geotag, T , can be reproduced, such that $T' = T$. The effective geotag length is $k \log N$.

Fuzzy Extractor Modified from Chang and Li's Secure Sketch

Unlike a fuzzy extractor, a secure sketch recovers the input at the calibration using a sketch. The main security requirement is that the published sketch, s , should not reveal essential information on the inputs; otherwise, it will be helpful to attackers.. Chang and Li proposed the small secure sketch for point set difference [10], which can be applied to geo-security systems with adequate modification. Location data that are continually contaminated by noise cannot be recovered exactly; therefore, the secure sketch must be modified to a fuzzy extractor. The distance measure is also a Hamming metric in this approach. The construction of the modified approach is as follows.

Calibration. Given that $q_x = \{q_1, \dots, q_n\}$,

1. Construct a monic polynomial, $p_1(x) = \prod_{i=1}^n (x - q_i)$.
2. Publish $P = \langle p_1(0), p_1(1), \dots, p_1(2t - 1) \rangle$.

Verification. Given $P = \langle p_1(0), p_1(1), \dots, p_1(2t - 1) \rangle$ and $q'_x = \{q'_1, \dots, q'_n\}$,

1. Construct a new polynomial, $q_1(x) = \prod_{i=1}^n (x - q'_i)$.
2. Compute $q_1(0), q_1(2), \dots, q_1(2t - 1)$.

3. Let $p_2 = x^t + \sum_{j=0}^{t-1} a_j x^j$ and $q_2(x) = x^t + \sum_{j=0}^{t-1} b_j x^j$ be polynomials of degree t . Construct the linear equations with a_j 's and b_j 's as unknowns, which satisfy $q_1(i)p_2(i) = p_1(i)q_2(i)$, for $0 \leq i \leq 2t - 1$.
4. Find one solution of the linear system.
5. Solve for the roots of the two polynomials, $p_2(x)$ and $q_2(x)$. Let the roots be x' and y' , respectively.
6. The recovered location parameter is $q'_x = (y \cup x') \setminus y'$.

Lemma 1 in Chang and Li's paper states that the entropy loss due to enrollment is at most $2t \log N$ when $x \cap \{0, \dots, 2t - 1\} = \emptyset$. The assumption is that x does not contain any element from $\{0, \dots, 2t - 1\}$.

5.2.4 Summary of the Proposed Fuzzy Extractors

The Euclidean metric fuzzy extractor is designed to adjust the errors introduced by the random noise and the seasonal biases. The RS, secret sharing-based, and secure sketch-based fuzzy extractors help to reproduce geotags when there is an insufficient number of the location-dependent parameters or operating transmitters.

Since random noise and biases are always present in RF signals, a Euclidean fuzzy extractor should constantly be applied to minimize the impact of signal temporal variations and guarantee the reproducibility of geotags. Unlike noise and biases, errors due to missing parameters are infrequent. Users can choose when to use these Hamming metric fuzzy extractors. A combined use of Euclidean metric and Hamming metric fuzzy extractors can achieve more robustness in the derived geotags, with a tradeoff of more entropy loss.

5.3 Reproducibility Analysis

In this section, the performance of the first three fuzzy extractor constructions is examined and compared. The evaluation is based on the user's FRR, the attacker's FAR, and the geotag entropy loss.

All of the three constructions improve the consistency of location-dependent parameters, thus reducing the FRR. The users' false reject is affected by the variations in the parameters, the selected quantization step Δ , and the quantization offset, that is, how far the received parameters differ from the center of the quantization grid. The most desired scenario is when the distribution of the parameter is exactly in the middle of the quantization grid (offset = 0), whereas the worst-case scenario is when the distribution lies on the boundary of the grid (offset = 0.5Δ), depicted in Figure 5.5.



Figure 5.5: Quantization scenarios: best (left); worst(right)

Euclidean Metric Fuzzy Extractor

This section first examines how the reproducibility of a geotag improves using the Euclidean metric fuzzy extractor. The analysis is illustrated in Figure 5.6. The x -axis is the quantization steps in terms of the standard deviations σ ; the y -axis is the estimated FRR. The geotag is computed from TD, SS, and ECD using the seasonal data from four West Coast stations. As a result, there are 11 different location-dependent parameters.

To estimate FRR, the first day of the 90-day data is used to compute a geotag for calibration; the data from the remaining 89 days are used for the verification. The experimental FRR is the number of data samples in the 89 days, in which the geotags are matched with the one computed on Day 1, divided by the total data samples. The result indicates that after applying the Euclidean metric fuzzy extractor the estimated FRR is reduced by 84% from 0.433 to 0.066 when $\Delta = 4\sigma$.

From the mathematical analysis, the Euclidean metric fuzzy extractor rounds off the measurements at the verification step to the measurements at the calibration step. A geotag can be reproduced when the offset between the two measurements is less

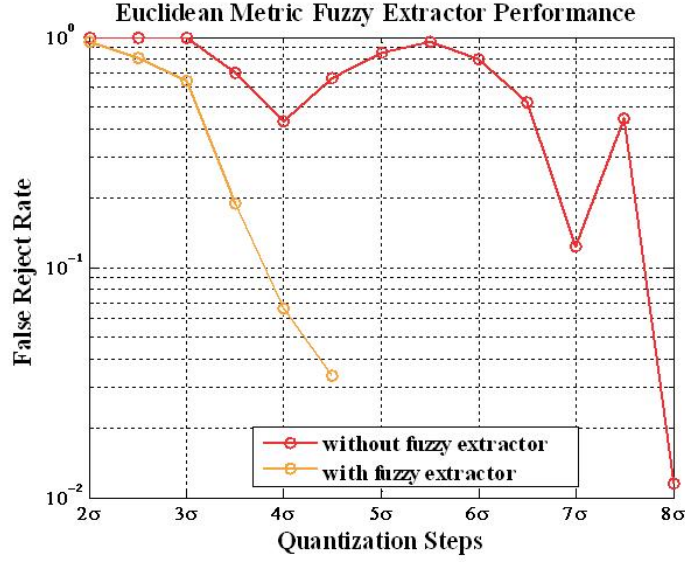


Figure 5.6: Euclidean metric FE performance improvement

than a threshold; that is, $T' = T$, if $\text{dis}(x, x') < \Delta/2$.

Reed-Solomon Error Correcting Review

This section provides a short review of Reed-Solomon codes, since the FRR of multiple parameters using the Hamming metric fuzzy extractor depends on the RS error-correcting performance. Let q be the code alphabet, that is, $q = 2b$, the number of possible symbols, where b is the number of binary bits in a symbol. RS codes are non-binary codes. The decoding algorithm of RS codes is defined as a bounded distance decoder; that is, only received sequences within a fixed-designed bound of a valid codeword can be decoded, and no errors can be corrected over the bound [15]. The representation of the decoder operation is illustrated in Figure 5.7. Each white dot corresponds to an actual RS codeword. The black dots enclosed by the circles are the possible received sequences that can be mapped to the closest codeword.

The decoder can make two types of errors: the received sequence is decoded in an incorrect codeword, called an *undetected error*; or, the received sequence is not decoded to any of the codewords, and is considered to be a *decode failure*. Let the purple circle in Figure 5.7 represent the correct codeword. If the received sequence

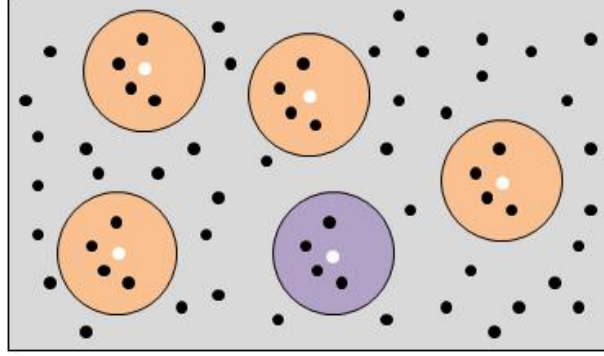


Figure 5.7: RS decoder representation

is within any other orange circle, the output codeword is incorrect, and it represents an undetected error. If the received sequence is in the gray region and not bounded by the circles, it is considered to be a failure. For an $RS(n, k)$, the minimum spacing between different codewords is $n - k$. The decoder can correct errors up to $t \leq \lfloor \frac{n-k}{2} \rfloor$. Let p be the symbol error or the error rate of one location-dependent parameter. The probabilities of incorrect decoding [15] are

$$Pr\{\text{error or decode failure}\} = \sum_{i=t+1}^n \binom{n}{i} p^i (1-p)^{n-i} \quad (5.11)$$

$$Pr\{\text{undetected error}\} = \frac{(q^k - 1) \sum_{i=0}^t \binom{n}{i} (q-1)^i}{q^n}. \quad (5.12)$$

The evaluation of error detections can be classified into three different scenarios to compute three probabilities [9]: that the decoded codeword is correct, P_C ; that the decoded codeword is an undetected error, P_{UE} ; and, that it fails to decode, P_F . For an $RS(n, k)$ code, the minimum distance between codewords can be defined as $d_{min} = n - k + 1$. Let u represent the number of symbol errors, and $0 \leq u \leq n$. The three probabilities can be computed as follows:

For $0 \leq u \leq t$, $P_C = 1$, $P_{UE} = 0$ and $P_F = 0$ since the decoder can correct error up to t .

For $t \leq u \leq d_{min} - t$, $P_C = 0$, $P_{UE} = 0$ and $P_F = 1$ since the received sequence

is too far from any of the possible codewords, and is thus considered to be a decode failure.

For $u \leq d_{min} - t$, $P_C = 1$, and $P_F = 1 - P_{UE}$. The undetected error probability can be computed using Equation (5.12).

RS-Based Hamming Metric Fuzzy Extractor

Multiple parameters can be used to achieve better robustness and security strength in a geotag. More location-dependent parameters provide more information entropy, better resolution, and increase the difficulty of predicting a target geotag. However, one drawback is that the system reliability is degraded. The reproducibility comparison with and without a Hamming metric fuzzy extractor is illustrated in Figure 5.8. Both cases use the Euclidean metric fuzzy extractor to minimize quantization errors and ensure that the measurements are in the middle of the quantization grids. The analysis uses 15 parameters to estimate the FRR; thus, $n = 15$. The overall system FRR using a Euclidean metric fuzzy extractor can be estimated as $1 - \prod_{i=1}^n (1 - p_i)$, where p_i is the error rate of one parameter or symbol error. For the combination of fuzzy extractors, the overall FRR, depicted in the orange color, is estimated using Equation (5.11). The number of errors, t , is selected to be 2, which results in a corresponding $k = 11$. The solid lines represent the analytical analysis, while the dots are estimated using the seasonal data collected at Stanford University. When the symbol error, p , is 0.1, the FRR is reduced by 86% from 0.4 to 0.055.

Secret Sharing-Based Hamming Metric Fuzzy Extractor

This section uses the Wi-Fi data illustrated in Figure 4.9 to evaluate the performance of the secret sharing-based Hamming metric fuzzy extractor. For simplicity, only APs are used to derive a geotag. Other location-dependent parameters such as the received signal strength (RSS) and the response rate can also be used for generating the Wi-Fi geotag. If RSS is one of the input parameters, the Euclidean metric fuzzy extractor should be applied. The performance analysis is illustrated in Figure 5.9. The blue line indicates the FRR of the derived geotag without using any fuzzy extractor, while the

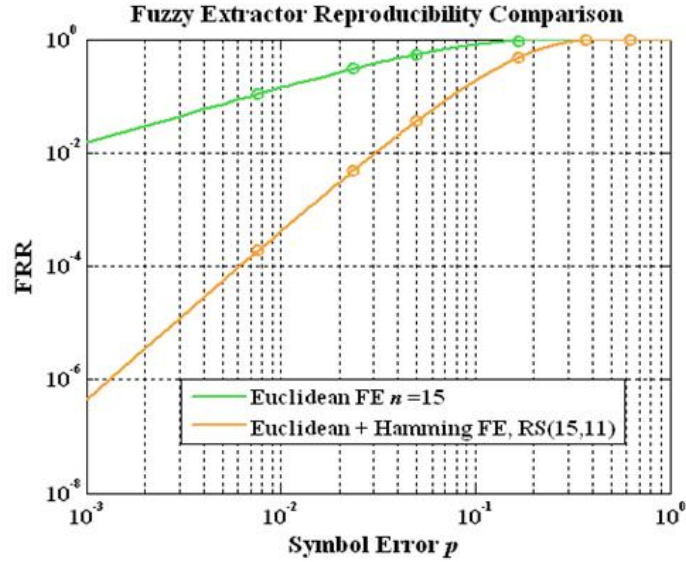


Figure 5.8: Performance of RS-based fuzzy extractor

red line represents the improved FRR after using the secret sharing-based Hamming metric fuzzy extractor. Figure 4.9 illustrates that only the strongest AP achieves a 100% response rate. It is expected that the geotags computed from two or more APs yield a low FRR or low reproducibility, as illustrated in Figure 5.9. The analysis also shows that the FRR is reduced by 85% when four APs are used to derive a geotag. The geotag still yields a high FRR when the number of APs is greater than four even with the secret sharing-based fuzzy extractor. Wi-Fi geotag reproducibility is location-dependent, since the coverage of Wi-Fi APs is different from one location to another.

5.4 False Accept Rate for Security Analysis

An important measure of security in a geo-security system is the false accept rate, which is the probability of an attacker acquiring a desired geotag successfully in white-listing applications. In this section, the effect of fuzzy extractors on security performance in a location-based security system is investigated.

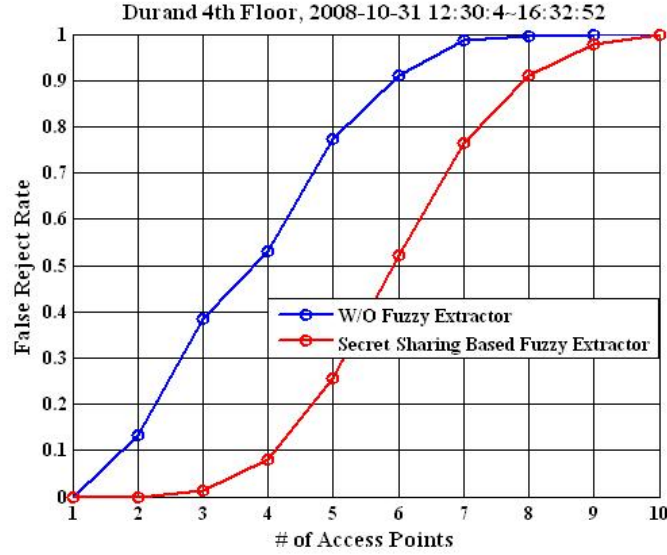


Figure 5.9: Performance of SS-based fuzzy extractor

Euclidean Metric Fuzzy Extractor

The FAR values are determined by the distance between the physical locations of an authorized user and an attacker, the variance and decorrelation of the location-dependent parameters, and the quantization steps selected by a user.

The Euclidean metric fuzzy extractor does not always increase or reduce an attacker's false accept rate. The key idea of this fuzzy extractor is to round off the user's measurements to a specific quantization level. For all the measurements, x' , if $\text{dis}(x - x') < \frac{\Delta}{2}$, the quantized parameter, q_x , can be recovered. This claim is also valid for attackers. For any attacker whose measurements are within $\frac{\Delta}{2}$ distance from x , the measurements can map onto a correct quantization grid. The following diagram explains the claim using two scenarios.

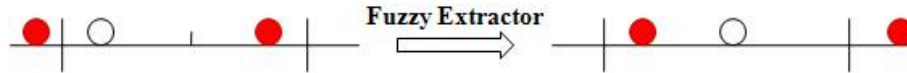


Figure 5.10: Security performance of Euclidean metric fuzzy extractor

Let the white ball in Figure 5.10 represent an authorized user, and the two red

balls represent two attackers. The two long vertical lines bound a quantization grid with a step Δ . The absolute distance between the white ball and the center of the grid is the public value, P . After applying the fuzzy extractor, the white ball is shifted to the center, and the left red ball is also moved within the grid, whereas the other red ball is moved outside. This diagram shows that the attacker's probability of a false accept depends on the closeness of the user and attacker. In addition, the more parameters used to compute a geotag, the higher the difficulty for attackers to break into the system.

RS-Based Hamming Metric Fuzzy Extractor

The false accept rate using an RS-based fuzzy extractor depends on the RS decoder performance. As mentioned, the decoder relies on a bounded distance to map into the closest codeword, and the bound is determined by n and k . The FAR is considered to be a type of undetected error. From an authorized user's point of view, an undetected error means that the received sequence is mapped into any one of the incorrect codewords, which are represented by the orange circles in Figure 5.7. From an attacker's point of view, an undetected error means that the received sequence is mapped into the user's codeword indicated by the purple circle in Figure 5.7. The total FAR of multiple parameters using an RS-based fuzzy extractor should be analyzed under two conditions: $dis(q_x, q_y) \leq t$ and $dis(q_x, q_y) > t$, where q_x is the quantized location vector of an authorized user and q_y is the vector of an attacker. The FAR estimation is derived from Equation (5.12), illustrated as follows:

$$FAR_{RS} = \begin{cases} 1 & dis(q_x, q_y) \leq t \\ \frac{q^k \sum_{i=1}^t \binom{n}{i}}{q^n} & dis(q_x, q_y) > t \end{cases}. \quad (5.13)$$

When the Hamming distance between the user and attacker's vectors is less than, or equal to, t , both the user and the attacker can reproduce the desired codeword. However, when the Hamming distance is greater than t , the probability of error is

fixed and depends on n and t alone. For instance, with $n = 15$ and $t = 2$, the probability that an attacker receives a desired codeword is approximately 0.00185, which is fairly low.

SS-Based Hamming Metric Fuzzy Extractor

This FAR analysis of secret sharing and secure sketch-based fuzzy extractors is similar to that of the RS-based metric. A desired geotag can be achieved if the number of errors is less than a certain threshold. For the secret sharing-based fuzzy extractor, the threshold is $n - k$; for Chang and Li's approach, the threshold is t , which is a design parameter. If an attacker's location-dependent parameters possess more errors than the threshold, the false accept rate will be low.

5.5 Tradeoff Analysis

As mentioned, multiple location-dependent parameters provide more security strength in a derived geotag but reduce the reproducibility of the geotag and the reliability for authorized users. The security strength of each parameter is different in terms of the information entropy, the reproducibility, and the false accept rate.

This section first studies the tradeoff between the false reject rate of authorized users after applying the fuzzy extractors introduced in Section 5.2 and the false accept rate of attackers as the number of location-dependent parameters increases. Loran data and the Euclidean metric fuzzy extractor are used for this study. The analysis is illustrated on the top of Figure 5.11. The error rates are also limited by the selected quantization steps. The two orange curves represent the FARs with quantization steps of 3σ and 6σ , whereas the two green curves indicate the corresponding FRRs. The error rates of the 3σ quantization step are represented by the circle marker and the solid line. The triangle marker and dashed lines indicate the error rates of the 6σ step. The seven selected parameters are TD from George, Middletown, and Searchlight, and the signal strength from all four stations in GRI 9940. The error rates are calculated using the seasonal data shown in Figure 3.34. To estimate

FRR, the collected data are referred to as the authorized user's. The first day of the seasonal data are considered to be the calibration measurement, whereas data from the remaining days are the verification measurements. The percentage of time that the geotags at the calibration and verification steps match is calculated. On the other hand, to estimate the FAR, the collected data are referred to as the attackers, whose parameters are 1σ off from the authorized user's parameter values. This study evaluates the tradeoff between the user false reject rate and the attacker false accept rate by varying the number of location-dependent parameters and the quantization steps.

The same Loran data set is used for the second tradeoff study, illustrated on the lower left of Figure 5.11. The tradeoff analysis is between the geotag reproducibility and information entropy. More entropy in the location-dependent parameters can result in a longer geotag. A longer geotag means that it is more difficult to break by a brute-force attack. With a sufficiently long geotag, a brute-force attack is not a threat to the system.

The left axis is the FRR of an authorized user; the right axis is the sum of the information entropy from the location-dependent parameters. The quantization step used in this analysis is 6σ . The result illustrates that a 56-bit geotag can be achieved using seven parameters with this typical quantization step. The time required to break a 56-bit geotag with one operation per μsec is 1142 years. With a supercomputer that performs 106 operations per μsec , it takes only 10 hours to break a 56-bit geotag [57].

The system FRR and the geotag resolution can be traded off against each other by varying the number of parameters used to compute the geotag, illustrated on the bottom right of Figure 5.11. Wi-Fi data in the residential area are used in this analysis. The FRRs are estimated after applying the secret sharing-based fuzzy extractor. The separation between the calibrated locations plays an important role in determining a lower bound on geotag resolution. Technically, the geotag resolution is inversely proportional to the number of APs in computing a geotag. However, the orange curve that represents the geotag resolution stops decreasing when the number of APs approaches six. On the other hand, the system FRR increases dramatically as the number of APs goes to four. Thus, the loss, an increase in FRR, is greater than

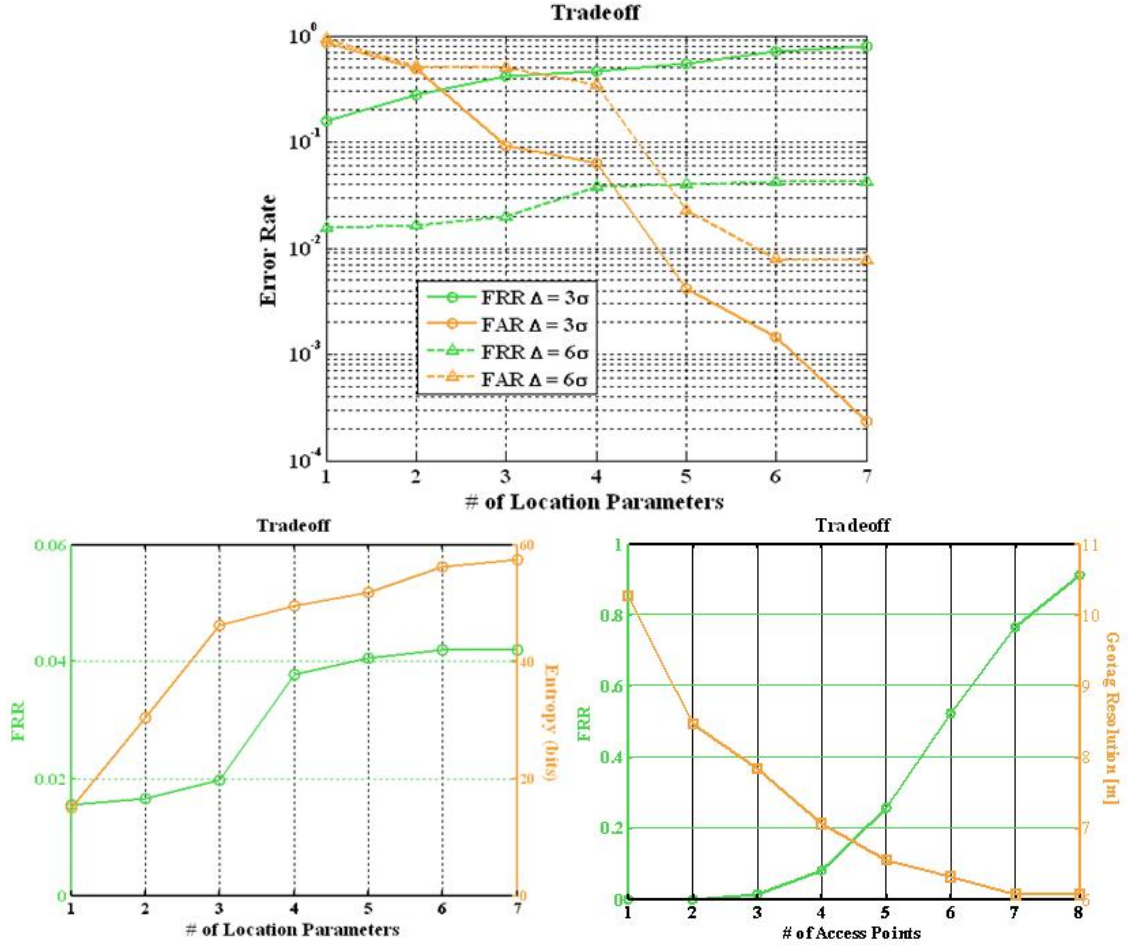


Figure 5.11: Trade spaces: FAR and FRR (left); FRR and entropy (middle); FRR and average cell diameter (right)

the gain, the reduction in the geotag resolution, when the number of APs is above four.

As a result, the FAR, FRR, information entropy, and geotag resolution can be properly designed based on the selected quantization steps and number of location parameters. A more secure system requires a smaller quantization step and larger number of parameters while a more convenient system prefers larger step and fewer parameters.

5.6 Conclusion

The Euclidean metric fuzzy extractor was developed for random noise, biases, and quantization errors to achieve high reproducibility for the derived geotag. Reed-Solomon-based, secret sharing-based, and secure sketch-based fuzzy extractors were designed for the scenario in which the RF transmitter is offline. The entropy loss of the Hamming metric fuzzy extractor is greater than the Euclidean metric fuzzy extractor. The algorithms for performance comparison of proposed fuzzy extractors are shown in Table 5.1, where $d^* = \text{dis}(q_x, q_y)$.

	FRR	FAR	Entropy Loss
Euclidean	$1 - \prod_{i=1}^n (1 - p_x)$	$\prod_{i=1}^n p_y$	$\sum_{i=1}^n \log \Delta_i$
RS-based	$\sum_{i=t+1}^n \binom{n}{i} p_x^i (1 - p_x)^{n-i}$	$\begin{cases} 1 & d^* \leq t \\ \frac{q^k \sum_{i=0}^t \binom{n}{i}}{q^n} & d^* > t \end{cases}$	$\sum_{i=1}^t \log N_i$
Secret Sharing	$1 - \prod_{i=1}^k (1 - p_x)$	$\prod_{i=1}^k p_y$	$\sum_{i=1}^{n-k} \log N_i$
Change and Li's	$\sum_{i=t+1}^n \binom{n}{i} p_x^i (1 - p_x)^{n-i}$	$\begin{cases} 1 & d^* \leq t \\ 0 & d^* > t \end{cases}$	$\sum_{i=1}^{2t} \log N_i$

Table 5.1: Fuzzy extractor algorithms for performance comparisons

Adequate quantization steps should be selected to achieve reasonable geotag reproducibility. The FAR, the FRR, and information entropy can be traded off against each other by varying the quantization steps and the number of location-dependent parameters. For example, a more secure system requires a smaller quantization step and a greater number of parameters, whereas a more convenient system is characterized by a larger step and fewer parameters. Users of location-based security services possess the flexibility to select appropriate design parameters based on their applications and performance requirements.

Chapter 6

Conclusions

6.1 Results and Contributions

This dissertation examines theoretical and practical aspects of geo-security or location-based security systems. A summary diagram of the system demonstration is illustrated in Figure 6.1. The results lead directly to the improved performance of conventional security systems by providing an additional layer of protection using location-dependent signal characteristics.

The contributions of these efforts include the following:

Theoretical Framework

Chapter 2 presents a detailed construction of a geo-security system, followed by the development of a standard process to evaluate system performance qualitatively and quantitatively. Specifically, two important performance standards are continuity and integrity, which are also used in the navigation field. Continuity refers to the reliability of a geo-security system in the presence of noise, biases, and practical implementation issues. Integrity indicates the security of a system and its ability to resist attacks; it is analyzed using a threat model, which includes different levels of attacks that threaten or weaken the system. The most common and easy-to-implement attack in a geo-security system is spoofing. A tamper-resistant device and self-authenticated

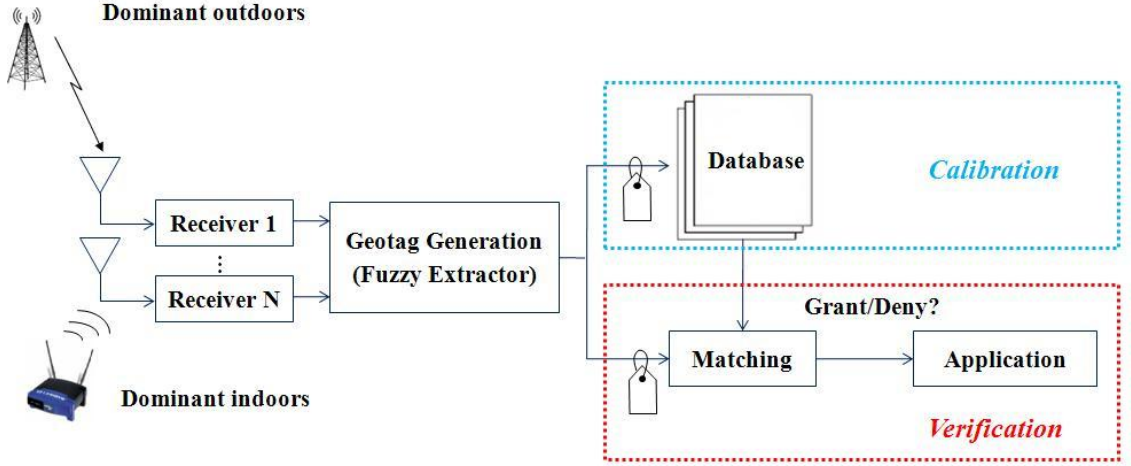


Figure 6.1: A robust geo-security system demonstration: 1) Multiple signals to improve spatial decorrelation and security strength of geotags; 2) Fuzzy extractors to tolerate temporal variations and reduce continuity risks, such that $T(t_1) = T(t_2)$.

signals preclude spoofing attacks. Second-level attacks include the two location-based attacks: a location brute-force attack (simple attack) and a selective-delay attack. In addition, a security system must be able to provide error bounds to all attacks under all conditions. The False Reject Rate (FRR) measures the consistency of location-dependent parameters, and thus, quantifies the continuity, whereas the False Accept Rate (FAR) bounds the probability of a simple location-based attack. The difficulty of the location-based attacks depends essentially on the spatial decorrelation of the location-dependent parameters used to compute a geotag. Furthermore, a methodology is developed to measure the location-dependent parameters from an information theoretical viewpoint. The trade space between continuity and integrity is quantified and demonstrated using a receiver operating curve by varying the parameter quantization steps.

System Validation Using Loran

Chapter 3 demonstrates and validates a geo-security system using Loran as a case study under the assumption that the device is tamper-resistant. A signal authentication scheme, TESLA, was proposed and implemented on Middletown, a Loran

transmitter from the West Coast chain GRI 9940. With the current LDC proposal and 50% data capacity available for authentication, the results from a data collection trip reveal that it takes at least 38.4 seconds to authenticate and verify the source of Loran signals.

A seasonal monitor station was set up at Stanford University. The 90-day seasonal monitor data were applied to evaluate the system continuity. Continuity risks due to the temporal variation of location-dependent parameters are observed with and without ASF mitigation.

To investigate the system integrity, a demonstration testbed was built for software and hardware. Various experiments were conducted to quantify the parameter spatial decorrelation, geotag resolution in different environments, and error bounds on the location-based attacks. A typical decorrelation distance of Middletown TD is 18 meters in an open-sky environment at Stanford. The result extrapolates that the false reject rate of attackers would be less than 0.01 when they are 18 meters away from a target user in such a scenario. In addition, the spatial data collected from a parking structure, soccer field, and office building illustrate that the minimum achievable geotag resolution is 9 meters. Furthermore, the error bounds on the location-based attacks indicated that a geo-security system using Loran can achieve high integrity.

System Validation Using Wi-Fi

Chapter 4 extends the analysis and performance evaluation using Wi-Fi together with Loran as a case study. Wi-Fi is chosen to complement Loran in indoor environments, since Loran signals are significantly attenuated when penetrating walls or buildings. Of the location-dependent parameters that can be used for a geotag computation, RSS and response rate yield larger temporal variations compared to MAC. One major continuity issue in Wi-Fi geotag generation is Access Point (AP) availability, that is, APs are not always observed and tracked by Wi-Fi devices. Furthermore, the AP availability or response rate is limited by the physical distance between the AP and the receiver. With the small signal coverage of Wi-Fi signals, the integration of Wi-Fi and Loran increases the spatial decorrelation and provides more information richness

or entropy to the geo-security system, thus lowering the success rate of location-based attacks. Experimental results demonstrated that geotag resolution was reduced from 9 meters to 3 meters in the Durand office building. With such a geotag-approached technique, no time synchronization is required between different RF systems on either the transmitters or the receivers.

Fuzzy Extractors

Chapter 5 develops four separate constructions of fuzzy extractors, which are error-tolerant algorithms to extract secret information from noisy location data. The Euclidean metric fuzzy extractor is designed for random noise, biases, and quantization errors; its performance is determined by the quantization step of the location-dependent parameters. The improvement of a Loran geotag computed from the 90-day seasonal data in false reject rates is 84% with a typical quantization step, 4σ . The Reed-Solomon (RS)-based, Secret Sharing (SS)-based and secure sketch-based Hamming metric fuzzy extractors should be applied when there are insufficient parameters or offline transmitters. The reduction in false reject rate is approximately 85% from 0.53 to 0.080 using Wi-Fi data collected in an office building with the secret sharing-based fuzzy extractor. These constructions differ in the sizes of the public information, decoding efficiency, and the degree of ease in practical implementation. The appropriate fuzzy extractor construction should be applied depending upon on the RF signals to implement geo-security, computation power, and the user's decision on the tradeoffs.

Fuzzy extractors can also improve the reproducibility of location fingerprinting or location signature for indoor positioning. The disadvantage of fuzzy extractors is the entropy loss in a derived geotag. The entropy loss can be traded off against geotag reproducibility by the number of errors to be tolerated, which is a design parameter.

6.2 Directions for Future Work

In this section, promising directions are presented for continued research investigation.

Signal Authentication Scheme for Navigation System

TESLA is a well-designed authentication scheme for communications; however, it provides only a modest level of security. With a spoofing module, there is still a chance for an attacker to modify the location-dependent characteristics and replay them to a target receiver. It would be helpful to design a new authentication scheme for navigation systems to make geo-security systems more robust to spoofing attacks.

Threat Model Update

A threat model is a very important step to analyze the security of a system by identifying relevant threats, vulnerabilities, and countermeasures to help shape the security design. This dissertation discusses six spoofing attacks and two location-based attacks. However, these may not be the only types of attacks that weaken a geo-security system. Therefore, it would be interesting to investigate other attacks to complete the threat model.

Signal Validation Using Other Signals

The beginning of Chapter 3 discussed possible signals that can be used to implement a geo-security system. Although this dissertation utilized Loran and Wi-Fi as case studies, other signals should be investigated and compared to determine the best signal for geo-security systems. If necessary, it would be interesting to design a new signal to meet the following requirements: anti-spoofing, high repeatable accuracy, high spatial decorrelation, and high information entropy of the location-dependent parameters.

Appendix A

Information Theory Review

This section presents some fundamental concepts of information theory. Information entropy, introduced by Claude Shannon more than a half century ago [14], is a measure of information density within a set of values with known occurrence probabilities. The information entropy of a discrete random variable X is defined by

$$H(X) = - \sum_{x \in X} p(x) \log p(x). \quad (\text{A.1})$$

The entropy of a finite measurement depends on the probability distribution of the random variable. The units for entropy are “nats” when natural logarithm is used and “bits” for base-2 logarithm [14]. This research uses base-2 logarithm rather than natural logarithm as base-2 logarithm provides more intuitive descriptions. If the probability distribution is uniform, the entropy can be represented as

$$H(X) = \log N. \quad (\text{A.2})$$

The uniform distribution provides the maximum information entropy for discrete random variables. The total number of occurrences is N , while the probability of each occurrence is $1/N$. It is worth mentioning that the normal distribution gives the maximum entropy for continuous random variables.

Another important definition in information theory is relative entropy or Kullback-Leibler divergence. Relative entropy, $D(P_X || P_Y)$, measures the difference between two

probability distributions, P_X and P_Y .

$$D(P_X||P_Y) = \sum_{x \in X} P_X(x) \log \frac{P_X(x)}{P_Y(x)}. \quad (\text{A.3})$$

Relative entropy is also a measure of the inefficiency of assuming that the distribution is P_Y when the true distribution is P_X . Since it is not symmetric, it is more a divergence measure than a distance measure, even though it has often been used as a distance metric [27].

The numerical estimation of entropy for finite data is fully discussed in [44]. Finite-size data introduces systematic errors that should be considered. The observation is that entropy not only fluctuates around its true value, but is also underestimated. The following is the corrected estimation.

$$H \approx H^{observed} + \frac{M-1}{N}. \quad (\text{A.4})$$

Here $H^{observed}$ denotes the observed entropy using a finite number of N data samples to estimate the probability of M discrete states.

Appendix B

Pattern Classification for Geotag Generation

Pattern classification [19] is the concept of assigning a physical object or measured data to one of the pre-specified groups, called *classes*, using *a priori* knowledge or statistical information. The patterns are the evaluated final decision from classifiers and represent the characteristics of *features*. Mathematical models are used as the theoretical basis for the classifier design. In classification, a pattern is referred to as a pair of variables $\{x, \omega\}$, where x is a collection of features or location-dependent parameters and ω is the concept associated with the features, also called *class label*.

The quality of location-dependent parameters or *features* is related to the ability to discriminate measurements from different classes. The goal is to maximize the differences between classes and minimize the inter-class scatter with the extracted decision rules from measurements, thus assigning class labels to future data samples.

Various classes of classification algorithms have been developed and successfully applied to a broad range of real-world domains. It is essential to ensure that the classification algorithm matches the properties of collected data in order to meet the needs of the particular applications. This dissertation selects three classifiers—linear discriminant analysis (LDA), k-nearest neighbor (kNN) and support vector machines (SVM)—to implement and generate a geotag.

B.1 Review of Classifiers

Linear Discriminant Analysis (LDA)

LDA is a traditional feature extraction method that aims for a transformation matrix that provides the optimal separation of multiple classes [19]. Data of all different classes are projected onto a subspace in which the data of different classes are as far apart as possible, whereas the data of the same classes are as close as possible. The optimal projection can be obtained by simultaneously minimizing the within-class scatter matrix norm and maximizing the between-class scatter matrix norm.

Fisher's linear discriminant (FLD) is the classical example of the linear classifier for two classes [58]. The between-class and within-class scatter matrices, S_B and S_w , are defined by

$$S_B = \frac{1}{M} \sum_{i=1}^c l_i (\mu_i - \mu_0)(\mu_i - \mu_0)^T, \quad (\text{B.1})$$

$$S_w = \frac{1}{M} \sum_{i=1}^c \sum_{j=1}^{l_i} (x_{ij} - \mu_i)(x_{ij} - \mu_i)^T, \quad (\text{B.2})$$

where x_{ij} indicates the j th training sample in class i , c is the number of classes, l_i denotes the number of training samples in class i , M is the total number of training samples, μ_i is the mean of the training samples in class i , and S_w denotes the covariance matrix of samples in class i .

The generalized Fisher criterion is defined by

$$J(W) = \frac{W^T S_B W}{W^T S_w W}, \quad (\text{B.3})$$

where w is the generalized eigenvectors of $S_B W = \lambda S_w W$ corresponding to d largest eigenvalues.

k-Nearest Neighbor (kNN)

The kNN classifier is a method for classifying data based on the distance or closeness to the training samples in the feature space. A similar idea for geotag generation was proposed in the previous study under the name nearest neighbor method (NNM) in [42].

The method relies on training samples about matching probabilities to consider the k -nearest neighbor rule [19]. The class labels are random variables and independent from each other; each has the probability of $P(\omega_i|x)$. The kNN rule selects ω_m with probability $P(\omega_m|x)$ if a majority of the k nearest neighbors have a label of ω_m . The value k is a design parameter, that is, the probability to select ω_m is larger if the value of k is greater. A large k reduces the impact of noise and produces smoother decision boundaries, but requires higher computation power. When $k = 1$, kNN becomes the nearest neighbor method.

Support Vector Machines (SVM)

SVM aims to minimize the structural risks. It not only classifies all the training samples correctly, but maximizes the margins between different classes. The problem of overfitting, which degrades the generalization ability, might occur while maximizing the classification performance. In this problem, high generalization ability results in a low false reject rate (FRR). By controlling model complexity, the simplest model that explains data is preferred to avoid overfitting [6].

Let M n -dimensional training samples, x , belong to two classes. With linearly separable data, the decision function, also referred to as the hyperplane, can be defined as

$$g(x) = w^T x + w_0, \quad (\text{B.4})$$

where w is an n -dimensional vector and w_0 is a bias term. The problem of deciding the optimal separating hyperplane can be formulated as

$$\begin{aligned}
& \text{minimize} && J(w) = \frac{1}{2}\|w\|^2, \\
& \text{subject to} && y_i(w^T x_i + w_0) \geq 1, \quad i = 1, 2, \dots, M.
\end{aligned} \tag{B.5}$$

If the training data are not linearly separable, the computed classifier may not have high generalization ability even with optimal separating hyperplanes. As a result, to enhance linear separability, the original data are mapped into a higher dimensional space in which data are more linearly separable.

While the SVM classifier maximizes the generalization ability, it is vulnerable to outliers due to the use of sum-of-square errors. Outliers should be mitigated before training to prevent their effects. A margin parameter C controls misclassification errors. A large value of C results in small hyperplane margin and good generalization ability, thus suppressing misclassification errors, whereas a small value of C results in large hyperplane margin and more misclassification errors.

B.2 Classifier-Based Geotag Generation

To develop an effective geo-security system using pattern classification, it is essential to acquire a thorough understanding of the input feature space and develop proper mapping of such feature space onto the output *classification space*. The proposed machine learning approach adopts representative statistical models to extract the characteristics of patterns in the feature domain. Different machine learning models should be selected based on the perspective of applications. Practically, the machine learning models have been adopted to construct a robust information processing system for other authentication systems, such as biometrics. The technique is potentially useful in a broad spectrum of application domains, including, but not limited to, geo-security.

The dimension of data is the number of random variables that are measured on each observation. A higher dimension of data, or more features to compute a geotag,

results in high spatial discrimination in a geo-security system, as well as total information entropy in a geotag. In practice, however, the added features may actually degrade the geotag reproducibility or reliability of the system, which significantly depends on the training sample size, the number of features for geotag generation, and the algorithm complexity. Such a phenomenon is referred to as the “curse of dimensionality.” Dimensionality reduction, which constructs a low-dimensional representation of high-dimensional data, is a means to avoid the curse of dimensionality and improve computational efficiency, classification performance, and the ease of modeling.

Both calibration and verification steps involve data collection, signal processing, feature extraction, dimensionality reduction, and classification. At calibration, a model is determined based on the training data. The model should be saved for future classification at the verification step. The geotag, T , associated with a location is obtained from the class label, ω , such that $T = \hbar(\omega)$, where $\hbar(\bullet)$ is a mapping function. All of the computed geotags will be stored in the database. At verification, the developed model is applied to classify the reduced dimension data; a new geotag is computed using the same mapping function from the extracted class labels. The matching algorithm to decide whether the computed geotag is authentic or not, is the same as the one for the quantization-based geotag matching.

B.2.1 Experimental Results

This section evaluates LDA, kNN, and SVM-based geotags in terms of spatial discrimination and geotag reproducibility using a Loran data set. A geotag with high spatial decorrelation ensures that users at different locations with small separation can achieve different geotags, thus lowering false accept rates (FARs).

The data set was collected at three test points in a parking structure at Stanford University to examine the three classifiers. A visualization of the three locations in green markers is shown in Figure B.1. The separation between test points 1 and 2 is approximately 70 m and the separation between test points 2 and 3 is around 30 m.

The location-dependent parameters – TD, ECD, and SNR – from four West Coast



Figure B.1: Three test locations in a parking structure at Stanford University

stations are used to derive a geotag. As a result, the input location feature vector is 11-dimensional. A linear dimensionality reduction algorithm is applied to lower the input vector dimension to two to achieve better spatial decorrelation.

LDA

The two-dimensional data $[x^1, x^2, x^3]$ that represent three locations are labeled Classes 1, 2, and 3 and plotted in Figure B.2. The estimated classifier is visualized as a separating surface, which is piecewise linear. The input data were trained using the Perceptron learning algorithm, which minimizes the distance of misclassified points to the decision boundary. The algorithm is an iterative procedure that builds a series of vectors $[w; w_0]$ until the inequality condition is satisfied. The inequality is represented as

$$[w; w_0] \cdot z_i^y > 0, \quad i = 1, \dots, 3, \quad (\text{B.6})$$

$$z_i^y(j) = \begin{cases} [x_i^T, 1]^T, & \text{for } j = y^i, \\ -[x_i^T, 1]^T, & \text{for } j = y^i, \\ 0, & \text{otherwise.} \end{cases} \quad (\text{B.7})$$

There is more than one solution when the input data are separable. The final solution depends on the initial vector $[w; w_0]^{(0)}$, which can be selected arbitrarily. The algorithm does not converge when the data are not separable.

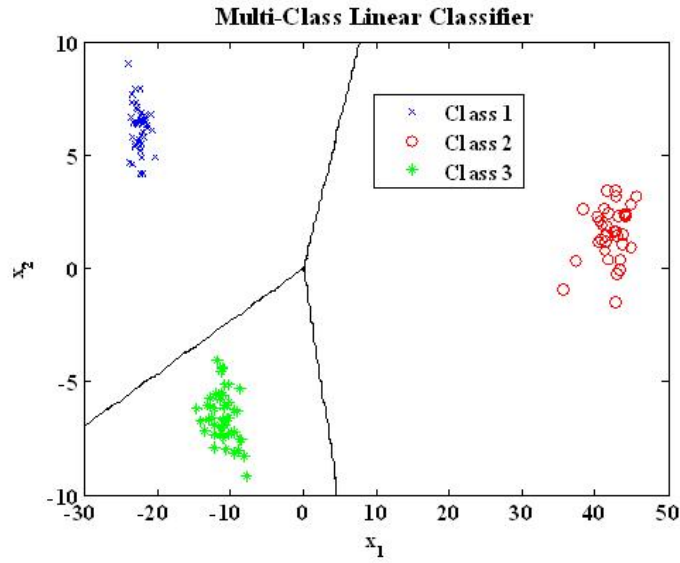
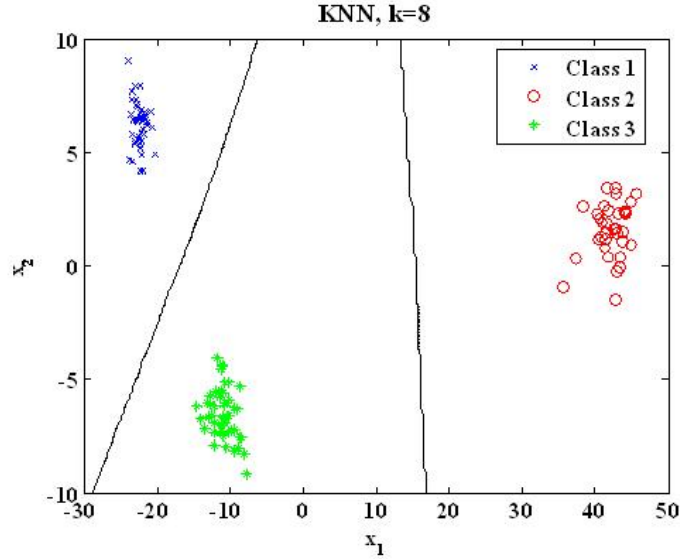


Figure B.2: Multi-class linear classifier trained by the Perceptron algorithm

kNN

The best choice of k depends on the input data; large values of k reduce the effect of the noise. The decision boundaries of the case $k = 8$ are plotted in Figure B.3. The algorithm is easy to implement but computationally intensive, especially when the training data size grows. Euclidean distance is used to measure the closeness between samples.

Figure B.3: The decision boundaries of kNN, $k = 8$

SVM

As mentioned earlier, SVM is considered as an optimization problem. To solve the optimization, Sequential Minimal Optimization (SMO) is applied. The One-Against-One (OAO) decomposition is used to train the SVM classifier. An input parameter, or kernel argument, controls the size of the hyperplane margin, thus adjusting the misclassification errors.

The 90-day seasonal monitor data were applied to examine the geotag reproducibility using the SVM classifier; the estimated FRR of the geotag is depicted in Figure B.5.

The FAR decreases as the kernel argument increases. Figure B.4 depicts the tradeoff between the margin and capacity by varying the kernel argument. When the kernel argument is small, the decision boundary is better fitted to the training data, thus raising the misclassification errors for future verification and decreasing system reliability. On the other hand, a large kernel argument results in fewer misclassification errors but increases the likelihood that an attacker can map into a correct geotag or FAR. An adequate kernel argument is the one with which both a low FAR and high spatial discrimination can be achieved.

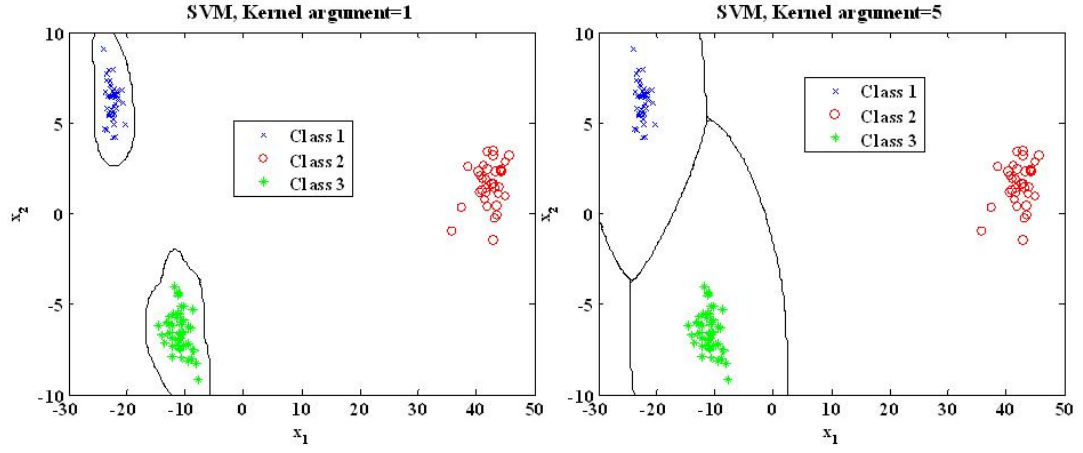


Figure B.4: Multi-class SVM classifier by OAO decomposition: kernel argument = 1 (left); kernel argument = 5 (right)

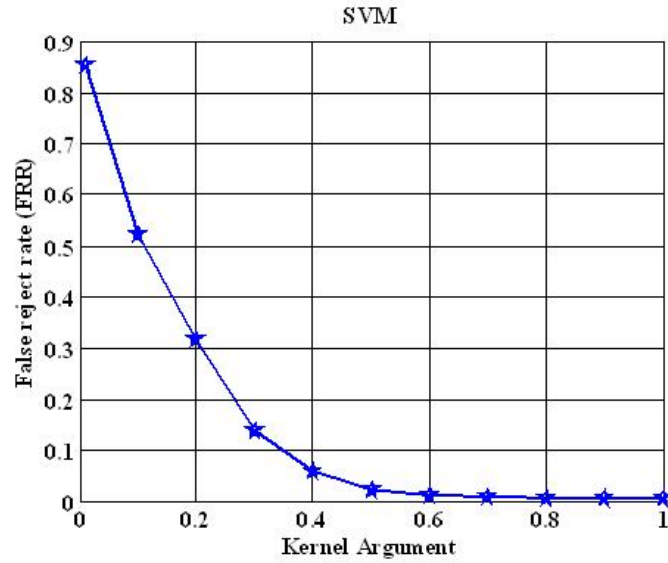


Figure B.5: FRR of SVM classifier-based geotag

In summary, classifier-based geotag generation algorithms are proposed to achieve high spatial discrimination, since the quantization-based method has the following limitations: 1) quantization introduces errors that degrade the system reliability; and 2) users should understand the training data in order to choose adequate quantization steps. The pattern classification uses machine learning techniques that improve not

only the spatial decorrelation of computed geotags but also users' convenience. The location data can be trained automatically based on three classifiers, LDA, kNN, and SVM.

Appendix C

Acronyms and Symbols

AES	Advanced Encryption Standard
AGPS	Assisted GPS
AOA	Angle of Arrival
AP	Access Point
ASF	Additional Secondary Factor
AWGN	Additive White Gaussian Noise
BPF	Band-pass Filter
bps	bits per second
CA	Certification Authority
CONUS	Conterminous United States
dB	decibels (logarithmic measurement of power)
dBm	decibels relative to one milliwatt
dBW	decibels relative to one watt
DMP	Digital Manners Policy
DoD	Department of Defense
DoS	Denial-of-Service
DSA	Digital Signature Algorithm
DSP	Digital Signal Processing
ECD	Envelop-to-Cycle Difference
E-field	Electric Field

eLoran	Enhanced Loran
ELR	Enhanced Loran Receiver
ELRR	Enhanced Loran Research Receiver
FAA	Federal Aviation Administration
FAR	False Accept Rate
FD	False Detection
FE	Fuzzy Extractor
FEC	Forward Error Correction
FLD	Fisher's Linear Discriminant
FRR	False Reject Rate
GHz	gigahertz (billions of cycles per second)
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GRI	Group Repetition Interval
H-field	Magnetic field
HMAC	key-Hash Message Authentication Code
IEEE	Institute of Electrical and Electronic Engineers
INS	Inertial Navigation System
ION	Institute of Navigation
kHz	kilohertz (thousands of cycles per second)
km	kilometer
kNN	k-Nearest Neighbor
L1	1575.42 MHz
L-band	All frequencies between 1 and 2 GHz
LDA	Linear Discriminant Analysis
LDC	Loran Data Channel
LOP	Line-of-Position
Loran	Long Range Navigation
LOS	Line-of-Sight
LPF	Low Pass Filter
MAC	Message Authentication Code

MAC	Medium Access Control
MD	Missed Detection
MHz	megahertz (millions of cycles per cycle)
mW	milliwatt
NEBW	Noise Equivalent Bandwidth
NPC	Ninth Pulse Communications
OA0	One-Against-One
PCI	Phase Code Interval
PDA	Personal Digital Assistant
PDF	Probability Density Function
PPRF	Physical Pseudo Random Function
PRF	Pseudo Random Function
PSD	Power Spectral Density
RF	Radio Frequency
RFI	Radio Frequency Interference
RFID	RF Identification
rms	root mean square
ROC	Receiver Operating Curve
RS	Reed-Solomon
RSA	Rivest, Shamir, and Adleman
RSS	Received Signal Strength
SMO	Sequential Minimal Optimization
SNR	Signal-to-Noise Ratio
SS	Secret Sharing
SS	Secure Sketch
SV	Space Vehicle
SVM	Support Vector Machines
SZC	Standard Zero Crossing
TESLA	Timed Efficient Stream Loss-tolerant Authentication
TD	Time Difference
TOA	Time-of-Arrival

TTA	Time-to-Alert
UHF	Ultra High Frequency
USCG	United States Coast Guard
Wi-Fi	Wireless Fidelity
WLAN	Wireless Local Area Network
WWII	World War II

Bibliography

- [1] What is tamper-resistant hardware? RSA Laboratories.
- [2] Loran-C user handbook. COMDTPUB P16562.6, Nov. 1992.
- [3] Loran-C signal specification. COMDTINST M15662.4A, Commandant, USCG, May 1994.
- [4] The cost of movie piracy. Motion Picture Association of America (MPAA), 2005.
- [5] Password recovery speeds: how long will your password stand up. LockDown, Jan. 2007.
- [6] S. Abe. *Support Vector Machines for Pattern Classification*. Springer-Verlag London Limited, 2005.
- [7] P. Bahl and V. N. Padmanabhan. RADAR: An in-building RF-based user location and tracking system. *IEEE infocom*, 2:775–784, Mar. 2000.
- [8] J. V. Boone. *A Brief History of Cryptology*. US Naval Institute Press, May 2005.
- [9] K. M. Carroll, A. Hawes, B. Peterson, K. Dykstra, P. Swaszek, and S. Lo. Differential Loran-C. In *Proceedings of ENC GNSS*, 2004.
- [10] E. Chang and Q. Li. Small secure sketch for point-set difference. *Cryptology ePrint Archive*, 2005.
- [11] E. C. Chang and Q. Li. Hiding secret points amidst chaff. *Eurocrypt*, 4004:59–72, 2006.

- [12] E. C. Chang, R. Shen, and F. W. Teo. Secure sketch for multi-set difference. *Cryptology ePrint Archive*, 2006.
- [13] Y. Cheng, Y. Chawathe, A. LaMarca, and J. Krumm. Accuracy characterization for metropolitan-scale WiFi localization. In *Proceedings of Mobisys*, 2005.
- [14] T. M. Cover and J. A. Thomas. *Elements of Information Theory 2nd Edition*. Wiley-Interscience, 2 edition, July 2006.
- [15] A. Daraiseh and C. Baum. Decoder error and failure probabilities for reed-solomon codes: Decodable vectors methods. *IEEE Trans. Commun.*, 46(7):857–859, July 1998.
- [16] D. E. Denning and P. F. MacDoran. Location-based authentication: Grounding cyberspace for better security. *Computer Fraud & Security*, pages 12–16, Feb. 1996.
- [17] D. E. Denning and L. Scott. Using GPS to enhance data security. *GPS World*, Apr. 2003.
- [18] Y. Dodis, L. Reyzin, and A. Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. *SIAM Journal on Computing*, 38(1):97–139, 2008.
- [19] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley-Interscience, second edition edition, 2000.
- [20] J. Hruska. Microsoft patent brings miss manners into the digital age. MSNBC, June 2008.
- [21] A. Juels and M. Sudan. A fuzzy vault scheme. *IEEE International Symposium on Information Theory*, 2002.
- [22] A. Juels and M. Wattenberg. A fuzzy commitment scheme. In *Proceedings of ACM Conference on Computer and Communication Security*, pages 28–36, 1999.

- [23] K. Kaemarungsi. *Design of indoor positioning systems based on location fingerprinting technique*. PhD thesis, University of Pittsburgh, 2005.
- [24] D. V. Klein. Foiling the cracker: a survey of, and improvements to password security. In *Proceedings of 2nd USENIX Workshop Security*, pages 5–14, 1990.
- [25] D. E. Knuth. *Art of Computer Programming, Volume 1: Fundamental Algorithms, Third Edition*. Addison Wesley, 1997.
- [26] K. W. Kolodziej and J. Hjelm. *Local Positioning Systems: LBS Applications and Services*. CRC, 1 edition, May 2006.
- [27] S. Kullback. *Information Theory and Statistics*. Dover Publications, July 1997.
- [28] J. D. Lasica. The prince of darknet. Darknet, May 2005.
- [29] S. Lo. *Broadcasting GPS integrity information using Loran-C*. PhD thesis, Stanford University, 2002.
- [30] S. Lo, R. Wenzel, G. Johnson, and P. Enge. Assessment of the methodology for bounding loran temporal ASF for aviation. In *Proceedings of ION NTM*, Jan. 2008.
- [31] S. C. Lo, B. B. Peterson, P. K. Enge, and P. Swaszek. Loran data modulation: extension and examples. *IEEE Transactions on Aerospace and Electronic Systems*, 43(2):628–644, Apr. 2007.
- [32] P. Misra and P. Enge. *Global Positioning System: Signals, Measurements and Performance*. Ganga-Jamuna Pr, Dec. 2001.
- [33] A. Perrig, R. Canetti, J. Tygar, and D. Song. The TESLA broadcast authentication protocol. *CryptoBytes*, 5(2):2–13, Summer/Fall 2002.
- [34] O. Pozzobon, C. Wullems, and K. Kubik. Requirements for enhancing trust, security and integrity of GNSS location services. In *Proceedings of ION Annual Meeting*, pages 65–75, 2004.

- [35] D. Qiu. Security analysis of geoencryption: A case study using Loran. In *Proceedings of ION GNSS*, Sept. 2007.
- [36] D. Qiu, D. Boneh, S. Lo, and P. Enge. Robust geotag generation from noisy location data for security applications. In *Proceedings of ION ITM*, Jan. 2009.
- [37] D. Qiu, S. Lo, and P. Enge. Geoencryption with Loran. In *Proceedings of International Loran Association*, Oct. 2007.
- [38] D. Qiu, S. Lo, and P. Enge. Security for insecure times: Geoencryption with Loran. *GPS World*, pages 35–40, Nov. 2007.
- [39] D. Qiu, S. Lo, and P. Enge. A measure of loran location-based information. In *Proceedings of ION PLANS*, Apr. 2008.
- [40] D. Qiu, S. Lo, P. Enge, and D. Boneh. Pattern classification for geotag generation. In *Proceedings of ION GNSS*, Sept. 2009.
- [41] D. Qiu, S. Lo, P. Enge, D. Boneh, and B. Peterson. Geoencryption using Loran. In *Proceedings of ION NTM*, Jan. 2007.
- [42] D. Qiu, D. D. Lorenzo, S. Lo, and P. Enge. Physical pseudo random function in radio frequency sources for security. In *Proceedings of ION ITM*, Jan. 2009.
- [43] T. Roos, P. Myllymaki, H. Tirri, P. Misikangas, and J. Sievanen. A probabilistic approach to WLAN user location estimation. *International Journal of Wireless Information Networks*, 9(3):155–164, July 2002.
- [44] M. S. Roulston. Estimating the errors on measured entropy and mutual information. *Physica D*, 125(3-4):285–294, Jan. 1999.
- [45] J. Ryder. Laptop security, part one: preventing laptop theft. SecurityFocus, July 2001.
- [46] S. Saha, K. Chaudhuri, D. Sanghi, and P. Bhagwat. Location determination of a mobile device using IEEE 802.11b access point signals. *IEEE Wireless Communications & Networking Conference (WCNC)*, 3:1987–1992, Mar. 2003.

- [47] B. Schneier. *Applied Cryptography: Protocols, Algorithms, and Source Code in C, Second Edition*. Wiley, 2nd edition, Oct. 1996.
- [48] B. Schneier. Kill switches and remote control. Schneier on Security: A blog covering security and security technology, July 2008.
- [49] L. Scott. Anti-spoofing & authenticated signal architectures for civil navigation systems. In *Proceedings of ION GNSS*, Sept. 2003.
- [50] L. Scott and D. Denning. A location based encryption technique and some of its applications. In *Proceedings of ION NTM*, Jan. 2003.
- [51] L. Scott and D. Denning. Location based encryption & its role in digital cinema distribution. In *Proceedings of ION GNSS*, Sept. 2007.
- [52] S. Singh. *The Code Book: The Science of Secrecy from Ancient Egypt to Quantum Cryptography*. Anchor, reprint edition, Aug. 2000.
- [53] D. R. Stinson. *Cryptography: Theory and Practice, Third Edition*. Chapman & Hall/CRC, 3 edition, Nov. 2005.
- [54] B. Sullian. The biggest data disaster ever. MSNBC, Nov. 2007.
- [55] P. Swaszek, G. Johnson, R. Hartnett, and S. Lo. Assessment of the methodology for bounding loran temporal ASF for aviation. In *Proceedings of ILA 36th Annual Meeting*, 2007.
- [56] U. Uludag, S. Pankanti, S. Prabhakar, and A. K. Jain. Biometric cryptosystems: issues and challenges. *Proceedings of IEEE*, 92(6):948–960, June 2004.
- [57] L. Williams. A discussion of the importance of key length in symmetric and asymmetric cryptography, 2002.
- [58] X. Wu, K. Wang, and D. Zhang. Palmprint recognition using fisher’s linear discriminant. *International Conference on Machine Learning and Cybernetics 2003*.

- [59] C. Wullems, M. Looi, and A. Clark. Signal authentication and integrity schemes for next generation global navigation satellite systems. In *Proceedings of ENC GNSS*, 2005.