UNIVERSITÀ DEGLI STUDI DI PADOVA

UNIVERSITÀ DI PADOVA FACOLTÀ DI INGEGNERIA

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

SCUOLA DI DOTTORATO IN INGEGNERIA DELL'INFORMAZIONE

CURRICULUM IN SCIENZA E TECNOLOGIA DELL'INFORMAZIONE

XXIX Ciclo

Authentication and Integrity Protection at Data and Physical layer for Critical Infrastructures

Dottorando

GIANLUCA CAPARRA

Supervisore:

Dr. Nicola Laurenti

Direttore della Scuola:

Ch.^{mo} Prof. Matteo Bertocco

Anno Accademico 2016/2017

Abstract

This thesis examines the authentication and the data integrity services in two prominent emerging contexts such as Global Navigation Satellite Systems (GNSS) and the Internet of Things (IoT), analyzing various techniques proposed in the literature and proposing novel methods.

GNSS, among which Global Positioning System (GPS) is the most widely used, provide affordable access to accurate positioning and timing with global coverage. There are several motivations to attack GNSS: from personal privacy reasons, to disrupting critical infrastructures for terrorist purposes.

The generation and transmission of spoofing signals either for research purpose or for actually mounting attacks has become easier in recent years with the increase of the computational power and with the availability on the market of Software Defined Radios (SDRs), general purpose radio devices that can be programmed to both receive and transmit RF signals.

In this thesis a security analysis of the main currently proposed data and signal level authentication mechanisms for GNSS is performed. A novel GNSS data level authentication scheme, SigAm, that combines the security of asymmetric cryptographic primitives with the performance of hash functions or symmetric key cryptographic primitives is proposed. Moreover, a generalization of GNSS signal layer security code estimation attacks and defenses is provided, improving their performance, and an autonomous anti-spoofing technique that exploits semi-codeless tracking techniques is introduced. Finally, physical layer authentication techniques for IoT are discussed, providing a trade-off between the performance of the authentication protocol and energy expenditure of the authentication process.

Contents

List of Figures

List of Tables

List of Acronyms

3GPP 3rd generation partnership project

ACF Auto-Correlation Function

ADC Analog to Digital Converter

AER Authentication Error Rate

AES Advanced Encryption Standard

AGC Automatic Gain Control

AOA Angle-Of-Arrival

APV Approach oPerations with Vertical guidance

AR Authentication Rate

ASIC Application Specific Integrated Circuit

AWGN Additive White Gaussian Noise

BER Bit Error Rate

BOC Binary Offset Carrier

BPF Band Pass Filter

BPSK Binary Phase-Shift Keying

C/A Coarse/Acquisition

C/N_0 Carrier to Noise Ratio

CBOC Composite BOC

CCDF Complementary CDF

CDF Cumulative Distribution Function

CDMA Code Division Multiple Access

CED Clock and Ephemeris Data

CIoT Cellular Internet of things

CNAV Civilian NAVigation

CS Commercial Service

CSCG circularly symmetric complex Gaussian

DES Data Encryption Standard

DLL Delay Lock Loop

DoS Denial of Service

DS Digital Signature

DSA Digital Signature Algorithm

DS-SS Direct Sequence Spread Spectrum

EC Elliptic Curve

ECDSA Elliptic Curve-DSA

EDBS External Data Broadcast Service

EGNOS European Geostationary Navigation Overlay System

ENISA European Network and Information Security Agency

ERIS External Region Integrity Service

ETSI European Telecommunication Standard Institute

FA False Alarm

FDMA Frequency Division Multiple Access

FEA Forward Estimation Attack

FEC Forward Error Correction

GLRT Generalized LRT

GNSS Global Navigation Satellite Systems

GPS Global Positioning System

GSM Global System for Mobile communications

GST Galileo System Time

I/NAV Integrity Navigation Message

ICAO International Civil Aviation Organization

ICD Interface Control Document

I-code Integrity Codes

i.i.d. independent and identically distributed

IMU Inertial Measurement Unit

IOD Issues Of Data

IoT Internet of Things

IRNSS Indian Regional Navigational Satellite System

IV Initialization Vector

JNR Jamming-to-Noise Ratio

K-L Kullback-Leibler

LDPC Low-Density Parity-Check

LFSR Linear Feedback Shift Registers

LLR Log-Likelihood Ratio

LNA Low Noise Amplifier

LOS Line-Of-Sight

LRT Likelihood Ratio Test

MAC Message Authentication Code

MAP Maximum *A Posteriori*

MBOC Multiplexed BOC

MD Missed Detection

MEDLL Multipath-Estimating Delay Lock Loop

MIMO Multiple-Input and Multiple-Output

ML Maximum Likelihood

MMSE Minimum Mean Square Error

MPKC Multivariate Public Key Cryptography

MSB Most Significant Bit

MTBA Mean Time Between Authentications

NF Noise Figure

NIST National Institute of Standards and Technology

NMA Navigation Message Authentication

NPA Non-Precision Approach

NTP Network Time Protocol

OMSD One out of Many Syndrome Decoding

OS Open Service

OTAR Over-The-Air Rekeying

pdf probability density function

PKI Public Key Infrastructure

PLE Path Loss Exponent

PLL Phase Lock Loop

pmd probability mass distribution

PMF probability mass function

PNT Position, Navigation and Timing

PRN Pseudo-Random Noise

PRS Public Regulated Service

PSD Power Spectral Density

PVT Position, Velocity and Time

QZSS Quasi-Zenith Satellite System

RAIM Receiver Autonomous Integrity Monitoring

RBS Rainbow-Band-Separation

ROC Receiver Operating Characteristic

SAS Signal Authentication Sequence

SBAS Satellite-Based Augmentation System

SCE Spreading Code Encryption

SCER Security Code Estimation and Replay

SD Syndrome Decoding

SDR Software Defined Radio

SEDLL Spoofing Estimating Delay Lock Loop

SHA Secure Hash Algorithm

SIS Signal in Space

SNR Signal-to-Noise Ratio

SoL Safety-of-Life

SSSC Spread Spectrum Security Codes

SV Space Vehicle

TBA Time Between Authentications

TESLA Timed Efficient Stream Loss-Tolerant Authentication

TEXBAT Texas Spoofing Test Battery

TLS Target Level of Security

TMBOC Time-Multiplexed BOC

ToA Time of Arrival

TTA Time To Alarm

TTFAF Time To First Authenticated Fix

UTC Coordinated Universal Time

VSA Vector Signal Analyzer

VSD Vestigial Signal Defense

WAAS Wide Area Augmentation System

WSAN Wireless Sensors and Actuators Networks

Chapter 1

Introduction

Cyber security is becoming a core task for all information systems. This is even more true for critical infrastructures on which the availability of essential services or the safety of the citizen depend. There exist different types of protection that achieve different goals. The main security services can be classified as:

- *authentication,* ensures that the information comes from the intended source,
- *data integrity,* ensures that the information is not maliciously altered,
- *secrecy,* protects the information from being disclosed to unintended users,
- *access control,* protects a resource from being used by unauthorized users.

It is noteworthy that security services are pursued separately and independently of each other and that in general, a mechanism providing a security service does not automatically provide others. A trivial example are encryption schemes, that provide confidentiality, but do not provide authentication. Indeed, the encryption scheme that provides perfect secrecy is the One Time Pad (OTP) that is the modulo-2 sum of the message with a perfectly random secret key of the same length. An attacker that does not know the secret key, observing the chipertext is not able to get any information neither on the original message nor on the secret key, but can alter the decoded message by simply changing any bit of the chipertext, e.g., performing a modulo-2 sum. The receiver will be able to decode the message but will not be able to detect that the message was altered, thus the mechanism does not provide integrity protection.

This thesis examines the authentication and the data integrity services in two prominent emerging context like Global Navigation Satellite Systems (GNSS) and the Internet of Things (IoT), analyzing various techniques proposed in the literature and proposing novel methods.

1.1 Motivation

The Global Positioning System (GPS), the most known GNSS, provides affordable access to accurate positioning and timing with global coverage. For this reason its usage has become widespread, from road transportation to maritime navigation. Even critical infrastructures such as telecommunication networks rely on it for time synchronization. In 2001 the Volpe report [1] assessed the vulnerability of infrastructures relying on GNSS, showing that GPS is vulnerable to both intentional and unintentional interference that may degrade or deny the Position, Navigation and Timing (PNT) service.

Since then, the navigation community have actively investigated the vulnerabilities of the radio navigation systems and have proposed countermeasures to increase the robustness both on the system

side, introducing security features in the signal, and on the receiver side, proposing signal processing techniques that make it harder to mount a spoofing attack.

In 2009 the European Commission, through its Joint Research Centre, investigated the hardening of GNSS tracking systems [2] in order to comply with the Commission Regulation (EC) No 2244/2003 on Vessel Monitoring System (VMS) that states: *Member States shall adopt the appropriate measures to ensure that the satellite-tracking devices do not permit the input or output of false positions and are not capable of being manually over-ridden, the master of a Community fishing vessel shall ensure that the satellite-tracking devices are fully operational at all times and prohibits to destroy, damage, render inoperative or otherwise interfere with the satellite tracking device.* Another mandatory application for GNSS tracking is the *digital tachograph*, defined by Commission Regulation (EC) No 165/2014.

The widespread adoption of GNSS trackers, for instance by delivery companies, in order to monitor their fleet, stimulates the market of low cost jammer, devices that broadcast an interference signal preventing GNSS receiver operation. Moreover, in recent years Software Defined Radios (SDRs), general purpose radio devices that can be programmed to both receive and transmit RF signals, have become available on the market easing the generation and transmission of spoofing signals both for research purpose or for actually mounting attacks.

For these reasons, GNSS security is an emerging topic that is actively investigated not only by the navigation community, but also by the system architects, to offer security service in GNSS in the near future. Indeed, the European Commission recently announced that by 2020 Galileo Open Service (OS) will provide data authentication.

The other context examined is IoT, a networking paradigm in which devices such as sensors and actuators are connected to the Internet to extend their functionality. This allows the construction of smart infrastructures, e.g., smart homes or smart cities, that collect and exchange data or that can be remotely controlled. A big part of IoT is encompassed by direct communication of devices, i.e., machine-to-machine (M2M). Being operated without human supervision and allowing remote control, a critical aspect is the security of these infrastructures. These devices are usually rather simple and inexpensive, with limited computation power and often battery powered. For this reason the security mechanisms shall be efficient and minimize their impact on the device. Unfortunately, traditional cryptography is in general not well suited for these constrained devices. A promising solution is physical layer authentication, in which the security mechanism is implemented directly at the physical layer, exploiting features of the channel without requiring heavy data exchange.

1.2 Contribution

The main contributions of this thesis are summarized in the following:

- A security analysis of the main currently proposed data level authentication mechanisms for GNSS, showing that the proposed key generation algorithm is not ideal. This analysis can be used to quantify the security loss with respect to an ideal key generation algorithm and, in turn, to dimension the system parameters to match the intended security level. The results on this topic were published in [3].

- The proposal of a novel GNSS data level authentication scheme that combines the security of asymmetric cryptographic primitives with the performance of hash functions or symmetric key cryptographic primitives. This data layer authentication scheme is designed to be flexible and can be used either standalone or in conjunction with a signal layer authentication mechanism. The results on this topic were published in [4, 5].

- The generalization of GNSS signal layer security code estimation attacks and defenses, improving their performance. This includes a new general class of estimators that can be optimized to

minimize the detection probability, and the introduction of a detection strategy that does not require the knowledge of the attack strategy. The results on this topic, both theoretical and experimental, were published in [6, 7].

- The introduction of an autonomous anti-spoofing technique that exploits semi-codeless tracking techniques. This technique exploits the higher chipping rate of military GNSS service to increase the complexity of the attacks, requiring more sophisticated hardware for the attacker, without requiring modification at system level and without requiring an aiding channel for the verification on the receiver side. The results on this topic were published in [7].
- The introduction of physical layer authentication techniques for IoT providing a trade-off between the performance of the authentication protocol and energy expenditure of the authentication process. The results on this topic were published in [8, 9].

1.3 Outline

The rest of the thesis is organized as follows:

- Chapter 2 introduces the fundamentals of radio navigation, the signals and the modulation; then provides a description of the receiver processing, highlighting how the design of the receiver may influence its security, and finally describes the major threats for GNSS.
- Chapter 3 discusses data layer authentication methods for GNSS, analyzing the currently proposed schemes and proposing a novel authentication scheme. Moreover, SBAS data authentication is discussed.
- Chapter 4 discusses signal layer authentication methods for GNSS, presenting results on both the attack and defense sides and the innovative usage of semi-codeless tracking techniques for autonomous anti-spoofing, and presenting an adaptation of integrity codes modulation to GNSS.
- Chapter 5 discusses physical layer authentication technique in the IoT context, presenting an energy aware node activation strategy.

Chapter 2

Satellite Navigation

2.1 Introduction to GNSS

Global Navigation Satellite Systems (GNSS) are radio-navigation systems that use satellites to broadcast navigation signals. The term *global* means that they provide coverage to the whole world, in opposition to regional navigation satellites systems. The development of GNSS started in the 1960s, with the goal of providing reliable and accurate positioning service. At that time both the US and the Soviet Union developed their own GNSS for military use, respectively GPS and GLONASS. In the 2000s Europe started the development of Galileo, her own GNSS. Also China is developing a GNSS, named BeiDou.

GNSS exploit Medium Earth Orbit (MEO) satellites, with orbital height around 20000 km, displaced in several orbital planes, that broadcast ranging signals. Each constellation counts between 20 and 30 Space Vehicles (SVs), so that from every point on the earth it is possible to see at least four SVs at every instant, see Fig. 2.1. From these ranging signals the receiver estimates the distance r between itself and the SV, measuring the propagation time T_p

$$r = cT_p = c(T_r - T_t) \tag{2.1}$$

where c is the speed of light and T_r and T_t are respectively the time of reception and transmission of the signal. This measurement is referred as *pseudo-range*, ρ, because it is not the true geometrical range r. Indeed, the receiver clock is not synchronized to the system clock, thus the reception time will itself contain an uncertainty term, δt_r, so T_r becomes

$$T_r = T_{rs} + \delta t_r \tag{2.2}$$

where T_{rs} is the reception time computed using the system clock, and we can rewrite the pseudorange as

$$\rho = c(T_{rs} + \delta t_r - T_t) = r + c\delta t_r \tag{2.3}$$

In order to compute his position, a receiver has to determine its three coordinates (x_r, y_r, z_r) and the time offset δt_r. This can be done by solving the non linear system of equations in the form of :

$$\rho_i = \sqrt{(x_i - x_r)^2 + (y_i - y_r)^2 + (z_i - z_r)^2} + c\delta t_r \tag{2.4}$$

where (x_i, y_i, z_i) are the coordinates of the i-th SV. The user shall solve the system in four unknowns, thus he needs to measure at least four pseudoranges at the same time. The system of equation can be solved using several techniques such as linearization or Kalman filtering. Once the receiver is able to get a position fix, it can adjust his internal clock synchronizing it to the GNSS system time. For this reason GNSS does not only provide positioning service, but also precise timing service,

allowing to get the accuracy of an atomic clock even on devices that run cheap local oscillators. It is worth noting that the Position, Velocity and Time (PVT) computation explained is rather simplified and the pseudoranges are affected by several other error sources, from the accuracy in the ranging measurement, to the multipath, the atmospheric effects and the errors in the SVs ephemeris and clock. Moreover, also the geometry of the constellation seen by the receiver impacts the position accuracy, since SVs with spatial diversity provide better accuracy than SVs that are closer together. The latter is a common situation in urban environments, where tall buildings block the reception of signals from low elevation satellites, allowing the reception only from a portion of the sky.

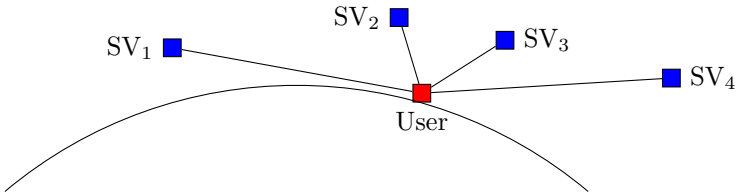


Figure 2.1: GNSS positioning principle.

2.2 GNSS signals

With the Galileo and BeiDou constellations completions, there will be four global navigation satellite systems, accompanied by regional navigation satellite systems, such as the Indian Regional Navigational Satellite System (IRNSS) and the Japanese Quasi-Zenith Satellite System (QZSS), plus Satellite-Based Augmentation System (SBAS), such as the European Geostationary Navigation Overlay System (EGNOS) and the Wide Area Augmentation System (WAAS).

GLONASS makes use of Frequency Division Multiple Access (FDMA), while the other GNSS make use of Code Division Multiple Access (CDMA). Each SV broadcasts multiple signals on two/three different frequencies. In order to not interfere each other, each signal is broadcast using a different spreading code, referred to as Pseudo-Random Noise (PRN). This allows a receiver to distinguish the signals coming from different SVs and to process them independently. This process, called Direct Sequence Spread Spectrum (DS-SS) spreads the power of the original narrow band signal over a much larger bandwidth, leading to a lower Power Spectral Density (PSD) and reducing the interference to other systems. Another advantage of this process is that it brings resilience to narrow-band interference. Indeed, if a narrow-band interferer term is added to the signal, when the receiver multiplies the received signal by the spreading sequence to recover the original signal, this will bring back the original PSD of the signal, while spreading the interferer noise on the wide band reducing the detrimental in band interference power. This forces an attacker to use a wide-band jammer in order to effectively disrupt the service, and the longer the spreading code, the wider shall the jammer bandwidth be.

On the other hand, due to both the distance of the satellites and the spreading operation, the received spectral density is very low, usually below the thermal noise. Thus, any further reduction of the received power, for instance due to shadowing effects of the environments, can prevent the use of the signal. Moreover, any very weak wide-band interference can easily impair the received signal acquisition.

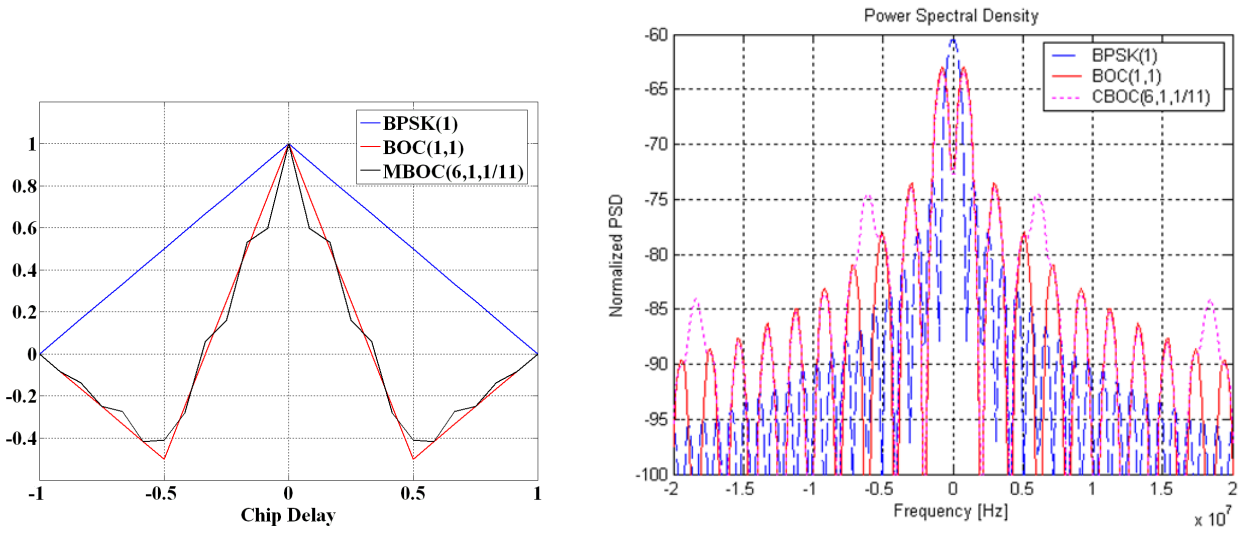
Among the GNSS signals, this thesis will focus mainly on Galileo E1, GPS Coarse/Acquisition (C/A) and GPS Precision (P) signals. The two GPS signals mentioned are Binary Phase-Shift Keying (BPSK) modulated, respectively BPSK(1) and BPSK(10), where the notation BPSK(n) indicate the relative chipping rate, respectively of 1.023 Mchip/s and 10.23 Mchip/s, and are modulated in quadrature on the same carrier frequency of 1575.42 MHz. The data rate is of 50 bps and no Forward Error Correction (FEC) is used. For C/A the spreading sequence is 1 ms long (1023 chip) thus each

data bit is represented by the 20 PRN repetitions. The GPS precision service instead uses a week long PRN sequence. The long spreading code, together with the higher chipping rate, allows much better pseudorange estimation. When working in anti-spoofing (A/S) mode, the GPS P signal is substituted by the P(Y) signal obtained by the modulo-2 sum with an encryption code, commonly referred to as W, with chipping rate 20 times lower than the P chipping rate (511,5 kchip/s). The two GPS signals are modulated in quadrature on the same carrier frequency of 1575.42 MHz.

The Galileo E1 signal uses a different modulation, named Composite BOC (CBOC), that is obtained by the composition of two square sub-carriers, i.e., Binary Offset Carrier (BOC). The selected modulation for E1 is a CBOC(6,1,1/11), meaning that the two sub-carriers, one of 1.023 Mchip/s, i.e., BOC(1,1), and one at 6.138 Mchip/s, i.e., BOC(6,1), are multiplexed with a power allocation of 10/11 to the lower chipping rate sub-carrier and 1/11 to the wider bandwidth sub-carrier. Moreover, the E1 signal is composed by the multiplexing of three signal component: E1A, the Public Regulated Service (PRS) service; E1B, the OS data component; and E1C, the OS pilot component, which does not carry any navigation message and is used to increase the tracking performance. The E1B navigation message is protected against channel impairments by using a convolutional coding with rate 1/2. The PRN is 4 ms long (4096 chip) thus each data symbol is represented by a single PRN repetition. Galileo E1 signal is modulated on the same carrier frequency of GPS L1, 1575.42 MHz, allowing interoperability among the systems.

SBAS signals are BPSK modulated with a data rate of 250 bps protected by the same convolutional coding as Galielo E1B, resulting in a 500 symbols per seconds stream. The PRN length is the same as GPS C/A, 1023 chip.

The different Auto-Correlation Functions (ACFs) and PSDs for the discussed modulation are shown in Fig. 2.2 [10]. It can be seen that the addition of high frequency components to the signal leads to a sharper ACF and introduce side-lobes in the PSD. These effects have two advantages: the sharper ACF allows better ranging performance and increase multipath rejection, while the spectral separation among the legacy signals that use BPSK and the modernized one that uses CBOC, or more generally other Multiplexed BOC (MBOC) implementations such as the GPS L1C that uses Time-Multiplexed BOC (TMBOC), allows to selectively deny the access to any constellation or service, without affecting the others.



(a) (b)

Figure 2.2: ACF (a) and PSD (b) for BPSK(1) used in GPS C/A, BOC(1,1) and CBOC(6,1,1/11) used in Galileo E1 OS [10].

Another important characteristic of a ranging signal is the chipping rate and the length of the spreading code. A higher chipping rate provides better ranging accuracy and increases the multipath resilience, while a longer spreading code improves the robustness to interference and reduce the cross-correlation. The drawback of the combination of a long spreading code transmitted at high chipping rate is that it results in a difficult signal acquisition. Indeed, the two legacy GPS signals, C/A and P, were designed to work in conjunction: the C/A signal was designed to help receivers easily acquire a low precision signal, estimating the Doppler frequency, the code delay and decoding the navigation data needed to generate the local replica of the P signal. At this point the receiver could perform an acquisition with a reduced search space. An advantage of increasing the chipping rate is that it makes harder to mount synchronized spoofing attack where the spoofing signal reaches the receiver's antenna synchronized with the legitimate signal to capture the tracking loop without causing cycle-slips.

Among the GNSS signals cited, only the military services, Galileo PRS and GPS P(Y), are designed with security features, while the civilian services are not. The security used in the military services is of the Spreading Code Encryption (SCE) type, where a non-public spreading code is used. The use of SCE means that an unauthorized user cannot exploit the processing gain to recover the signal; therefore, forgery of the signal (i.e. generating a "legitimate" spoofed signal) becomes intractable. For example, an attacker would need steerable dish antennas to get sufficient gain to be able to recover signals without knowledge of the encryption keys.

Instead, in open service signals an attacker can leverage the known spreading code and the predictable message structure to forge synthetic GNSS signals that will be deemed authentic by the receiver. In this thesis we will discuss system level techniques that improve the reliability of the ranging signals, by authenticating the navigation message (Chapter 3) and by increasing the complexity of forging a ranging signal (Chapter 4).

2.3 Receiver processing

A simplified GNSS receiver block diagram is reported in Fig. 2.3, while for a detailed description the reader is referred to [11].

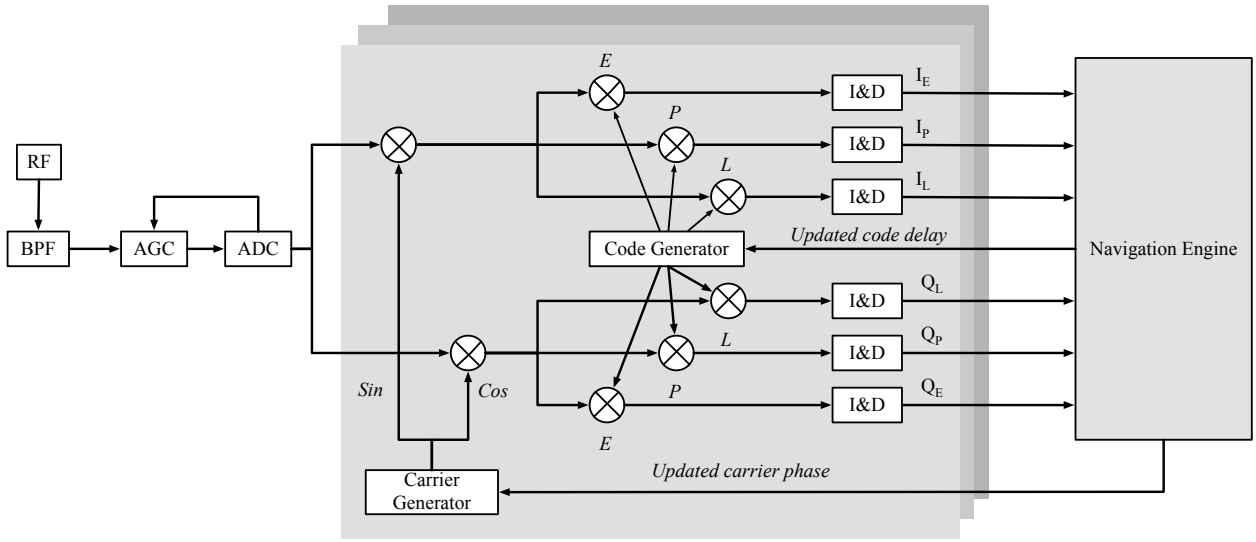


Figure 2.3: Simplified GNSS receiver block diagram.

The received RF signal is first down-converted to IF and filtered with a Band Pass Filter (BPF). If the device has multiple antennas or an antenna array it is possible to estimate the Angle-Of-Arrival (AOA) and check that the each signal comes from the correct direction [12]. If the attacker uses a single antenna to broadcast the spoofing signals, they will arrive all from the same direction, while authentic signals come from different directions. This may not be true in challenging environment such as urban canyons where most of the received signals come from multipath reflections and not Line-Of-Sight (LOS). On the other hand an antenna array is not practical for portable devices.

The bandwidth of the filter shall be designed based on the type of GNSS signals that the receiver is intended to process, i.e., Galileo E1 OS requires a wider bandwidth with respect to GPS C/A, and determine the amount of useful power that the receiver can collect for a signal, i.e., the number of side-lobes that the receiver is able to observe. In Fig. 2.2b it is possible to see that the CBOC(6,1,1/11) PSD presents the side-lobes corresponding to both the BOC(1,1) and the BOC(6,1). Receiver manufacturer can trade-off the ranging accuracy and robustness with the computational and hardware complexity. For instance mass-market receivers may process only the BOC(1,1) component, while professional receivers may process the full CBOC signal. Therefore, the bandwidth limitation shall be taken into account in the design of authentication techniques operating at signal layer.

The signal is then processed by an Analog to Digital Converter (ADC). The quantization depth of the ADC is another degree of freedom in the receiver design. Mass-market receivers use a limited number of bits, i.e., 1-3, in order to reduce the signal processing burden. This is justified by the limited degradation to the Signal-to-Noise Ratio (SNR) introduced by the quantization: 3.5 dB for 1-bit, 1.2 dB for 2-bit and 0.6 dB for 3-bit [11]. A non-cryptographic anti-spoofing technique called Vestigial Signal Defense (VSD) [13] exploits the fact that, assuming that the attacker it is not able to block the RF signal, the legitimate signal is still present even under a spoofing attack. VSD aims at detecting the simultaneous presence of both authentic and spoofed signal, through the monitoring of the complex correlation function. In order to have the resolution needed to detect weaker legitimate signal, 3-bit ADC might not be sufficient and thus for effectively implementing VSD a higher resolution is required. The main limitation of VSD is the multipath. Indeed, multipath corrupts the correlation function in a similar way to spoofing signals, thus VSD may results in high false alarm rate.

In order to keep the ADC in its dynamic range a feedback loop is present that adjusts the amplification factor used by the Automatic Gain Control (AGC). Usually the AGC sets its gain based on the received power and the noise level, but in the presence of interference it adjusts the gain in order to avoid saturation of the ADC. This effect can be exploited to monitor the signal integrity. Several works have investigated the use of AGC monitoring to detect spoofing attacks [14, 15]. The concept exploits the fact that, apart from environmental effects, the received power from the SVs changes slowly over time, thus sudden changes in the AGC level may indicate the presence of an interference signal. This control falls in the non-cryptographic integrity check class. Another control in this class is the Jamming-to-Noise Ratio (JNR) meter, that detects the presence and estimates the power, of a jamming signal. These controls are simple checks, easy to implement, that by themselves do not guarantee the authenticity of the signal processed, but can effectively limit the freedom for an attacker. For this reason some works assume the presence of this check, in order to avoid situations where the anti-spoofing technique alone is not effective, for instance the work presented in [16].

After the ADC, the digital signal enters into several tracking channels, each processing a different PRN. First, the estimated Doppler frequency and code phase are removed, then the samples are accumulated and the output is passed to the navigation engine. The accumulation can be a coherent integration, that offers optimal performance but whose duration must be limited mainly due to the receiver dynamics and the presence of navigation data. A way to extend the integration time is to use non-coherent integration which, however, suffers from squaring-losses. Another way to increase the integration time is by using semi-coherent integration [17, 18]. The correlation process exploits the processing gain of the DS-SS and increases the SNR of the output signal. For this reason it is

common to divide the digital processing in two stages: pre-correlation, where the signal is below the noise floor, and post-correlation, where the signal is above the thermal noise.

In order to track the received signal, a single correlator it is not sufficient. Various tracking loops were developed over years, many of them patented, thus a detailed description falls beyond the scope of the thesis. A general tracking loop employs six correlators, three for the In-phase (I) arm and three for the in-Quadrature (Q) arm. The three correlators are fed with the Early, Prompt and Late replica of the spreading code. A Delay Lock Loop (DLL), using the output of the correlators, corrects the code phase and code rate of the code generator, maintaining the Prompt replica aligned with the received signal. At the same time a Phase Lock Loop (PLL) corrects the carrier phase and carrier frequency of the carrier generator. The number of correlators and the spacing among them are degrees of freedom in the design of the tracking loops. Mass-market receivers may use a spacing of one chip between the Early and the Late replica in order to increase the tracking stability in low SNR conditions. On the other hand, professional receivers, that privilege the accuracy to the availability, tend to use narrower correlator spacing that result in better accuracy and multipath rejection.

In order to mitigate the mutipath, techniques operating at the correlator level such as Multipath-Estimating Delay Lock Loop (MEDLL) [19] were developed. MEDLL exploits a higher number of correlators in order to separate LOS and multipath signals, reducing the ranging error. A similar approach can be used to distinguish among the legitimate and spoofing signals with Spoofing Estimating Delay Lock Loop (SEDLL) [20]. Using more correlators allows also to implement Signal Quality Monitoring (SQM) methods that can be used to detect spoofing attacks [21].

It is worth observing that, somehow, the properties of the spreading sequences help the attacker in hiding the attack: two replicas of the signal correlate only if they are spaced of less than one chip. Indeed, the ACF of BPSK modulated signals have the triangular shape shown in Fig. 2.2a for less than one chip delay and remain close to zero for all the other delays. Moreover, the ACF of CBOC modulated signals fall to zero in less than 0.5 chip delay. This means that the correlation process itself suppresses a second signal if the relative delay falls in that zone. The higher resilience to multipath makes it harder for an attacker to capture the tracking loop, but at the same time increases the difficult of detecting a spoofing signal.

Apart from multipath, other effects may degrade the ranging performance, such as code-carrier divergence, originated from the ionospheric activity, and the code-subcarrier divergence [22], originated by the bandpass filtering of wideband signals such as high order BOC. Even for these phenomena there are techniques that mitigate the impact and work well against natural phenomena, but an exhaustive security assessment, that evaluates if an attacker can mimic the phenomena and exploit it, is missing.

If the receiver is equipped with an Inertial Measurement Unit (IMU), the measurements coming from this can be fused with the GNSS observables to improve the positioning accuracy. This is usually done by Kalman filtering, but the IMU can be used as input to the Kalman filter or can be used as an aid to the tracking loop, a technique known as *ultratight integration* [11]. The availability of the IMU allows also to crosscheck the GNSS dynamics with the reading coming from the inertial sensors. In this way the attacker is forced to maintain the coherence among the measurements, reducing its degrees of freedom. On the other hand good quality IMU are expensive and usually not adequate to portable devices. Based on the IMU grade, the attacker can exploit the uncertainty in the measurements to accumulate the desired position error over some time. It is worth noting that IMU are often calibrated using GNSS. In this case, the purpose of an independent cross check is lost.

Finally, moving to the navigation engine block, Receiver Autonomous Integrity Monitoring (RAIM) [23] is a technique developed to increase the reliability of the PVT. RAIM requires the tracking of more than four SVs in order to be able to compute multiple PVT excluding some ranging signal and checking if the resulting PVT solutions are consistent. If the inclusion of a signal leads to an inconsistency, the signal is discarded. The idea is to protect autonomous receivers from feared events and system failures, increasing the integrity of the PVT. Even in this case, the technique works well against random events,

but does not protect against spoofing. For example, assume that the receiver is tracking six SVs. If one of these signals is a spoofing signal, RAIM can effectively exclude it, protecting the receiver, but if the number of spoofed signals grows to four, then RAIM will exclude from the PVT computation the two authentic signals. Therefore, RAIM in this case favors the attacker. Indeed, without RAIM the attacker shall spoof all the six SVs in order to induce consistent PVT solutions.

For these reasons an anti-spoofing ranging signals and receiver processing shall be designed having in mind that a smart attacker will leverage any receiver processing techniques with the intention of mounting an attack, differently from the noise and environmental effects.

2.4 Augmentation systems

SBAS are radio navigation systems developed to work jointly with GNSS in order to increase the performance of the latter and to add integrity information in order to support Safety-of-Life (SoL) services such as aviation, maritime or railway.

SBAS design was heavily dictated by SoL services support, which posed many constraints in the design of systems, both on the ground segment side and on the Signal in Space (SIS). The most stringent requirements come from the aviation community. As example, Fig. 2.4 shows the fault-tree developed by International Civil Aviation Organization (ICAO) for approaches with vertical guidance (Approach oPerations with Vertical guidance (APV) I, II and category-I) [24]. It can be seen that, in order to achieve the Target Level of Security (TLS), an integrity risk of $2 \cdot 10^{-7}$ per approach is apportioned to the radio navigation system.
Some useful definitions from [25] are:

Integrity measure of the trust which can be placed in the correctness of the information supplied by the total system. Integrity includes the ability of a system to provide timely and valid warnings to the user (alerts).

Time To Alarm (TTA) is the maximum allowable time elapsed from the onset of the navigation system being out of tolerance until the equipment enunciates the alert.

Continuity of service is the capability of a system to perform its function without unscheduled interruptions during the intended operation.

Availability is the percentage of time over any 24-hour interval during which the predicted 95 percent positioning error (due to space and control segment errors) is less than its threshold, for any point within the coverage area. It is based on a 36-metre horizontal 95 percent threshold; a 77-metre vertical 95 percent threshold; using a representative receiver; and operating within the coverage area over any 24-hour interval. The service availability assumes the worst combination of two satellites out of service.

Table 2.1 gives an overview of the ICAO requirements for radio navigation aids [25] for each phase of flight such as En-route, Non-Precision Approach (NPA) and APV.

The reader may notice that these requirements are extremely challenging. GNSS and SBAS are designed to meet those requirements and are certified without authentication. Thus, any authentication protocol shall have virtually no impact on the performance and safety of the systems in order to maintain the certifications. On the other side, this gives the feeling on how much the radio-navigation aids are deemed reliable and how strongly some communities rely on them. This is in contrast with the complete lack of security features of current civilian services and with the growing awareness on spoofing attacks.

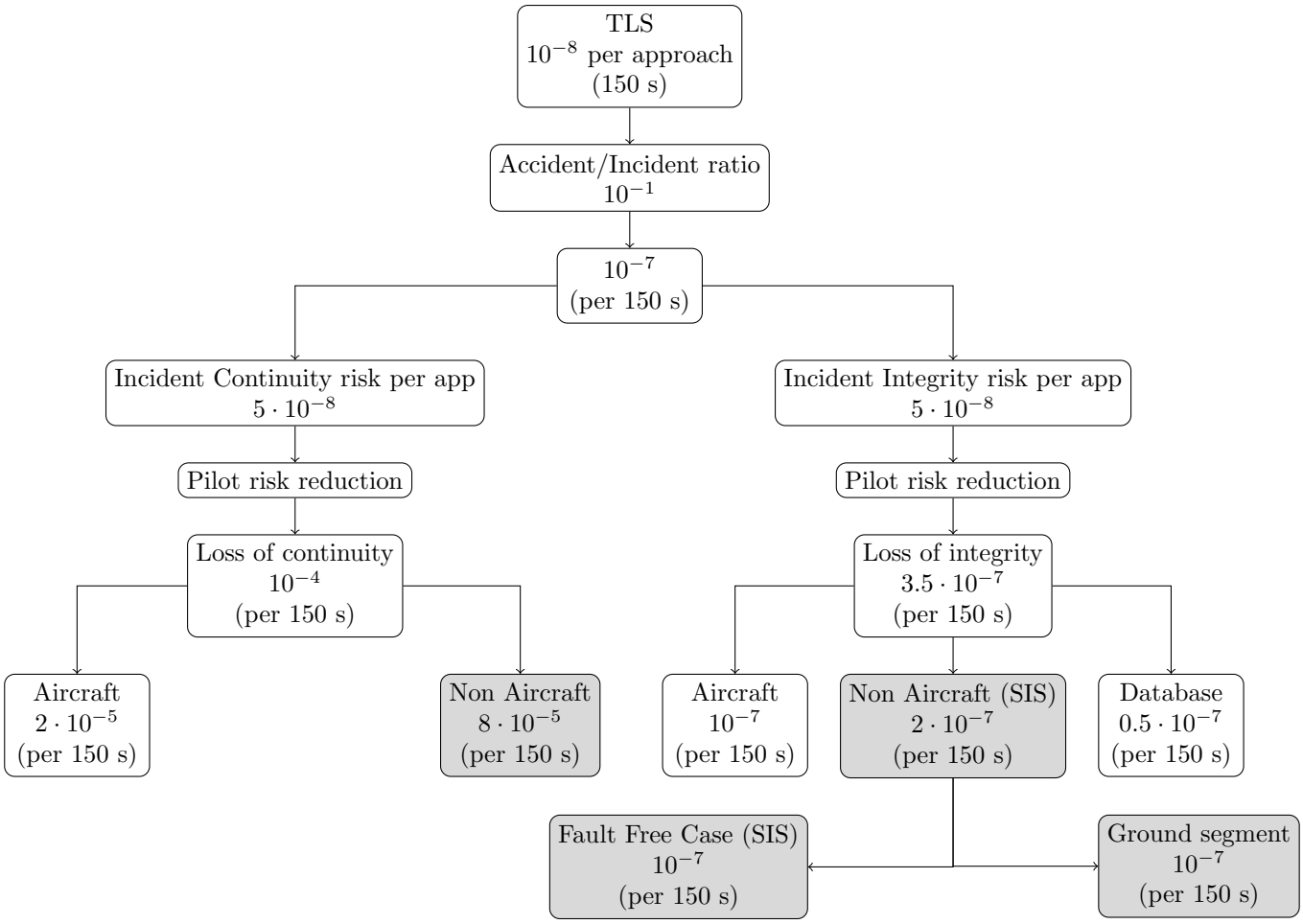


Figure 2.4: Fault-tree allocation for SBAS APV I, II and Category-I operations.

2.5 Attacks against GNSS

The information GNSS uses for PVT computation can be divided in:

- *navigation data:* that are the orbital parameters needed to compute the position of the SV at the time of transmission and its clock offset
- *propagation time:* an estimation of the Time of Arrival (ToA) is used to perform an estimation of the distance between the SV and the receiver. This estimation is usually referred to as pseudorange.

The attacks to the navigation function can be summarized as:

- *Data level attacks,* with the aim of inducing the receiver to use an incorrect navigation message in the PVT computation. This could be carried out by:
  - *Data forging or modification:* the attacker could generate arbitrary navigation data or modify the original ones, or
  - *Data replay:* the attacker could re-transmit old deprecated data with the purpose of intentionally degrading the output of the navigation function. In information security this is known as a replay attack.

Typical Operation	TTA	Integrity	Horizontal accuracy (95%)	Vertical accuracy (95%)	Availability	Continuity
En-route	5 min	$1-10^{-7}$/h	3.7 km	N/A	>0.99	$>$ $1-10^{-4}$/h
En-route, Terminal	15 s	$1-10^{-7}$/h	0.74 km	N/A	>0.99	$>$ $1-10^{-4}$/h
Initial & Intermediate approach, NPA, Departure	10 s	$1-10^{-7}$/h	220 m	N/A	>0.99	$>$ $1-10^{-4}$/h
APV I	10 s	$1-2\cdot10^{-7}$ /app	16 m	20 m	>0.99	$1-8\cdot10^{-6}$ per 15 s
APV II	6 s	$1-2\cdot10^{-7}$ /app	16 m	8 m	>0.99	$1-8\cdot10^{-6}$ per 15 s
CAT I	6 s	$1-2\cdot10^{-7}$ /app	16 m	<6 m	>0.99	$1-8\cdot10^{-6}$ per 15 s

Table 2.1: ICAO Signal in Space performance requirement [25].

  - *Data level DoS:* the attacker could maliciously modify the navigation message in order to prevent the correct receiver operation, mounting a Denial of Service (DoS) attack.

- *Ranging level attacks,* with the aim of inducing the receiver to use wrong ranging measurements in the PVT computation. This could be carried out by:
  - *Signal forging:* the attacker could generate arbitrary navigation signals,
  - *Signal relay:* the attacker could record and rebroadcast the entire RF signal modulated with unchanged code and data. In radar and satellite navigation domain this attack is known as meaconing, and is the analogous of the wormhole attack for ground wireless protocols.

Furthermore, it should be noted that the above classes of attacks can be combined. For instance, it is possible to forge a signal (i.e., spoofing the ranges) that conveys true navigation data (i.e., data replay). For a more comprehensive analysis on the vulnerabilities of GNSS, the readers can refer to [26, 27].

The problem of protecting the data level can be addressed with authentication techniques applicable to the information security domain, while the problem of protecting the ranging level belongs to the domain of signal or rather channel estimation, including also techniques such as signal watermarking. The protection of both levels is mandatory in order to provide assurance on the PVT, since both levels are used as input to the navigation function.

The mechanisms involved in the protection of the two layers are different and should pursue distinct design drivers. Although in principle it is possible to use a single technique to protect both layers, this may result in suboptimal performance. For instance, the adoption of Navigation Message Authentication (NMA) schemes can provide some assurance of ranging, based on the transmission of non-deterministic bit sequences; however, such protection is relatively weak when considered in the context of meaconing attacks with early bit prediction [28, 29] or Security Code Estimation

and Replay (SCER) [30, 6]. To improve the detection capability for these attacks, the number of unpredictable bits should be maximized. On the other hand, non-deterministic bit sequences can also have a negative impact on dissemination performance, particularly in challenging environments. Indeed, the navigation message is highly predictable for a given Issues Of Data (IOD); over time, receivers can accumulate the necessary pages of the navigation message containing ephemeris, clock correction terms, etc. With constantly changing bits of the message, the demodulation performance of authentication data are likely to degrade, affecting performance of NMA including error rate and availability of authentication.

In this thesis a *divide et impera* approach is followed, separating the mechanisms according to their main goal, in order to optimize each function and composing the results only at a later stage.

The main attack classes are:

Meaconing: all the signals are acquired by an attacker and later replayed to the receiver. This causes a time jump that is detectable if a trusted receiver is used as showed in [31].

Selective delay: an interferer, in order to avoid time jumps, could delay the signal that comes from a single SV only. Inducing a ranging error will affect both the computed time and position, by learning the position of the target user, it is possible to carefully design the attack in a way that the multiple induced ranging errors counteract, generating an error only in the position. Furthermore, if the attacker designs the spoofing so that it appears as a clock drifting, inducing the same time error for all the SVs, the receiver is not able to distinguish between clock drifting and a ranging error. However, this reduces his degree of freedom, and in a typical scenario, with many SVs in view, the position error can be reduced to a vertical range. Indeed, it would otherwise be possible to detect that the position is incorrect by using the RAIM algorithm [23], due to the inconsistency introduced by the replay.

Spoofing: an attacker can generate a signal with modified navigation data, e.g., the ephemeris, the clock correction parameters, the SV health indicator, or the GNSS to Coordinated Universal Time (UTC) time offset. Doing so he can introduce an error into the PVT solution computed by the target receiver.

Early bit detection: an interfering terminal can attempt to correlate a portion of the spreading code in order to detect the unpredictable bits, e.g., of the NMA [32], before reaching the end of the bit. If the bit detection is successful, it can be replayed and induce a negative delay to the pseudorange. This attack is discussed in [28] where it is shown that, for a 4092-chip code, a 1023-chip portion of the code could be detected and this could be used to introduce a negative delay in the pseudorange of up to 3 ms, which corresponds to an error of up to 900 km in ranging.

SCER: in the absence of authentication and integrity protection mechanisms, the known signal structure of the GPS L1 C/A signal and the predictability of its navigation data stream allows simple interfering techniques. Using more robust techniques at the code level for GNSS signal makes it more difficult to forge the signal due to the unpredictable secret code, although an interferer can still attempt to predict the signal trustful code and use it to create an artificial signal. This technique is much more sophisticated than simply ignoring the code or injecting a random value. The success of a SCER interference depends on the accuracy of the code estimate, which varies according to the estimation strategy and is limited by the code rate, because the accuracy of the chip estimates improves with increasing number of collected energy. SCER will be discussed extensively in Chapter 4.

Besides the GNSS vulnerabilities, an attacker could exploit the SBAS vulnerabilities to mount attacks against GNSS. Currently, SBAS L1 does not implement any message authentication feature and the signal characteristics are publicly known, so that an attacker can generate fake correction messages that

will be applied in the PVT computation. These messages can vary from fast ionospheric corrections, the impact of which is limited, to integrity messages that force the receiver to exclude measurements from faulty SVs.

Clearly, the attacks against GNSS and SBAS can be mixed. For instance the attacker can generate spoofed signals that carry correct navigation messages and NMA data, changing the ranging of lower elevation satellites, where it is harder to detect spoofing due to the low Carrier to Noise Ratio (C/N_0) and the multipath. Then, in order to fulfill integrity checks such as RAIM, it can send SBAS messages that exclude high elevation SVs from the PVT computation. For these reasons, SBAS are an important part of the GNSS infrastructure and shall be adequately protected in order to achieve resilient PNT.

Finally, if the attacker has physical access to the receiver, he can influence the PVT computation by tampering with the receiver. To prevent physical attacks a trusted, anti-tampering, receiver is required [31], where several monitoring functions check the hardware consistency of the receiver.

Chapter 3

Authentication at Data level

This Chapter will discuss authentication and integrity protection at the data layer both in GNSS and SBAS. The results on this topic were published in [3, 4, 5, 33].

3.1 Introduction to data level authentication

Navigation Message Authentication (NMA) aims to authenticate the origin and provide cryptographic integrity protection of navigation data to users. Navigation data is typically modulated on ranging signals at a low rate in order to minimize its impact on range estimation and provide adequate demodulation performances in a wide variety of environments for a message that changes infrequently. For example, the data rate of the Galileo OS dissemination channel is 120 bps; GPS C/A and L1C is 50 bps, the same bitrate as the Beidou D1 and GLONASS C/A signals. Therefore, NMA schemes are required to operate over a uni-directional broadcast channel and need to achieve an optimal tradeoff between:

- *Security:* maximizing robustness against attacks, including the choice of parameters such as: size of keys; security of algorithms; and security of key management functions such as key establishment.
- *Communications overhead:* minimizing the bandwidth requirements of NMA, i.e., the number of bits required for authentication, including the key management messages, e.g., renewal of the cryptographic keys.
- *Robustness to channel errors:* maximizing tolerance against errors in demodulation, especially in challenging environments.
- *Tolerance for data loss:* minimize impact of losing authentication data on continuity of operation, ability to recover from data loss.
- *Scalability of key management:* suitability of the scheme for large groups of users, particularly in relation to distribution and management of keys.
- *Computation and memory requirements of the receiver:* minimize the burden of NMA processing on the receiver.
- *Authentication performance:* maximizing performance including Time To First Authenticated Fix (TTFAF) and Authentication Error Rate (AER).

3.2 Review of broadcast authentication

Authenticating information transmitted over wireless broadcast channels is a common problem to many telecommunication applications (e.g., broadcast television) and a variety of solutions, usually referred to as *broadcast authentication* [34], have been proposed. An extensive classification and comparison of such schemes can be found in [35].

Digital Signature (DS) schemes appear to be an obvious choice for NMA, due to the simplicity and scalability of key management that come with the use of asymmetric cryptography. Many DS schemes such as those recommended by the European Network and Information Security Agency (ENISA) [36] (the suggested ones are RSA-PSS [37], RSA-DS2 [38], PV signature [39] and Schnorr signature [40] with the Elliptic Curve (EC) variant), are considered secure by the cryptographic community and a number of them additionally claimed to be provably secure. DS schemes often impose significant overheads on the user in terms of computational complexity and the size of keys and/or signatures. An option to reduce this overhead is to use elliptic curve variants of the cryptographic primitives, which are able to reduce both the signature and key size. For example, the traditional Digital Signature Algorithm (DSA) scheme requires a key of at least 1024 bits, whereas Elliptic Curve-DSA (ECDSA) requires a key size of just 160 bits for a security level of 80 bits. Both DSA schemes produce a signature of 320 bits; however, even 320-bits could be difficult to disseminate in tightly bandwidth constrained channels. For this reason, DS schemes are unlikely to be optimal for GNSS. Nevertheless, due to their cryptographical strength, several proposals of NMA schemes based on digital signature are present in the literature [32, 41, 42].

A possible solution to reduce the bandwidth is to use *digital signature schemes with message recovery* property, e.g., the Nyberg-Rueppel Signature [43]. The drawback of these class of schemes is that the message is embedded into the signature and thus the data demodulation requires verification of the signature, even if the receiver is not interested in authentication features.

Another alternative digital signature class that lead to short signature is based on the Weil pairings [44]. This digital signature scheme achieve a comparable security level to a 320-bit ECDSA signature with a signature size of just 154 bits. The disadvantage of this solution is that requires a computationally intense verification: in [44] is reported that a verification duration of 2.9 seconds on an Intel® Pentium III CPU clocked at 1 GHz, making it hardly suitable to low-end battery powered devices.

One time signature schemes, including BiBa [45] and HORS [46] are a class of DSs offering the advantage of fast verification at the cost of increased memory requirements. As a drawback, they can only authenticate a preset number of messages before renewal of the public key is needed. Therefore, the communication overhead and the public/private key size increase with the number of messages to be signed. For instance, if HORS is used to sign 10^4 messages with 80-bit security level, the required signature length is 200 B and the private and public key size are 24 MB and 48 MB, respectively [47].

An attractive alternative scheme that belongs to the *multi time signature* schemes is ETA [47], a variant of the Schnorr signature, where part of the information is offloaded from the signature itself to the public key. The author claims that this modification retains the provable security of the Schnorr signature. For the same 80-bit security level this yields a shorter 240-bit signature instead of 320-bit, with the drawback that the public key size increases with the number of messages that are to be authenticated with the same private/public key pair. If it is possible to update the public key through an aiding channel (e.g., a terrestrial network link once every several months or years) this scheme allows thus a further reduction on the bandwidth requirement.

Traditional broadcast authentication schemes based on symmetric cryptography are prone to compromise, as users must share the same secret key as the system. The secret key is used for both generation of an authentication code and its verification. Users of such schemes must be trusted to not forge messages, or stringent security requirements shall be imposed on the receiver such that key storage and cryptographic processing take place within a tamper-resistant hardware module.

Other broadcast authentication schemes utilizing symmetric cryptography attempt to mitigate the risk associated with key compromise through a delayed key disclosure paradigm. Each key that is used to generate an authentication code is disclosed to users after some delay, such that users only accept messages verified with the key if they have been received in a previous time window. One such scheme is Timed Efficient Stream Loss-Tolerant Authentication (TESLA), an authentication protocol on which several NMA schemes proposed in the literature [48, 49, 50, 51] have been based. A distinctive characteristic of TESLA is the use of a one-way key chain as a basis for delayed key disclosure. TESLA-based authentication schemes will be discussed in Section 3.2.2.

An alternative use of one-way chains will be proposed in Section 3.5, where digital signatures are amortized over a longer time period using a one-way chain of message digests. This approach potentially provides benefits both in terms of bandwidth efficiency and security.

3.2.1 Post-quantum cryptographic primitives

The security of all the well accepted and standardized digital signature protocols is based on difficult mathematical problems (e.g., the Integer Factorization or the Discrete Log Problem) over finite groups of integers numbers, or curve points. No polynomial time solution is known for any of these problems and it is conjectured that they are not solvable in a polynomial time, thus their parameters can be selected in order to be secure against a target computational power.

In the last thirty years the cryptographic community has warmly recommended the use of public key in general: every day a great amount of data is signed with (EC)DSA or encrypted via RSA, while Diffie-Hellman allows the secure distribution of the secret key materials.

However, in 1994, Peter Shor proved that if the classical computers will be substituted by quantum computers – i.e., that would work in accordance with the quantum mechanic principles – it will be possible to use quantum algorithms to efficiently solve difficult mathematical problems. This has been proved analytically by developing a quantum algorithm, the Shor's algorithm, which allows solving integer factorization and discrete logarithm in a polynomial time and with bounded error probability and can be used to break DSA and RSA-based cryptosystems. A second quantum algorithm worth mentioning is the Groove's algorithm, which provides a quadratic speedup over search algorithms. This speed up improves an hypothetical brute force attack on a quantum computer against any symmetric key scheme, which means that the key size shall be doubled in order to maintain the same security level against a quantum computer.

Even if it is not clear when large-scale quantum computers will become available, their potentialities threaten the security of many current cryptographic systems. Therefore, it is wise to be prepared to switch toward a quantum safe infrastructure. Since the last decade, the leading international organizations in security protocol standardization, such as the European Telecommunication Standard Institute (ETSI) [52] and the National Institute of Standards and Technology (NIST)[53] are actively discussing on the post-quantum topic. In accordance to their task, they are also evaluating the vulnerabilities of the well-known standards and primitives against the future technology. Each class of cryptographic primitives is affected as follows [53, 52]:

- Symmetric key algorithms without computational assumptions will continue guaranteeing unconditional secrecy (i.e., OTP) and security (i.e., Wegman-Carter), even against quantum attacks;
- Block ciphers, as AES, shall use longer keys;
- Hash functions, as SHA-2 and SHA-3, shall use longer output digests;
- Public key algorithms, as RSA, ECDSA, DSA, are expected to be no longer secure.

At the time of writing, there are multiple post-quantum primitives, meaning that no quantum algorithm able to break the underlying problem is known, yet. It is noteworthy that there is no proof

that such algorithms do not exist. Among the post-quantum schemes, two families are well suited for authentication purposes. Namely, *Error Correcting Codes* cryptography, and *Multivariate Polynomial* cryptography [54].

3.2.1.1 Code-based cryptography

This class of algorithms, based on error correcting codes, was originally introduced in 1978 by McEliece, to provide confidentiality. The ciphertext is a codeword randomly corrupted by errors, and the plaintext is retrieved by applying the error correcting code, as against channel errors. Its security is derived by the fact that it is difficult to decode such a ciphertext without any knowledge about the code applied at the transmitter side. It exists an equally secure variant of McEliece, the Niederreiter cryptosystem, that builds the ciphertext as syndrome, and consequently the plaintext is an error pattern instead of a codeword. These principles have never been broken. Many efforts have been done to deploy the McEliece algorithm as a digital signature scheme.

Courtois-Finiasz-Sendrier (CFS) signature The CFS algorithm [55] has been the pioneering work, which made it feasible to use code-based cryptosystem for signing purposes. Specifically, it uses the Niederreiter cryptoscheme with Goppa codes.

Let $\mathcal{C}$ be a binary linear code, which defines a space of 2^k codewords of n symbols. Let us define $m = \log_2 k$. The last code characteristic parameter is t, the error-correcting capacity of $\mathcal{C}$, that also defined the minimum distance between different codewords: $d = 2t + 1$. In public key cryptography approach, the secret key is a code $\mathcal{C}_0$ (usually a Goppa code) characterized by G_0 and H_0, while the public key is a code as well, but obtained by randomly permuting the coordinates of $\mathcal{C}_0$ through two non-singular matrices, U and V, and a permutation matrix P:

$$G = U \cdot G_0 \cdot P \quad \text{or} \quad H = V \cdot H_0 \cdot P \tag{3.1}$$

Therefore, the security is based on the principle that the decoding problem is difficult to solve, and so is that of retrieving the original G_0 and H_0 as well. The efficiency of the signing operation lies in finding the code parameters (i.e., n, k, t) such that the random hash result is likely to be a decodable syndrome. Indeed, the expected number of operations needed to target a syndrome is $t!$. On the other side, to counteract potential attacks with such a low error correcting capability, the codewords must be reasonably long. In [55] the CFS algorithm was designed to achieve a 80-bit security level with: $t = 9$, and $n = 2^{16}$ (m= 16). The resulting signature z is a sparse vector, that can be easily compressed to an average total length of 144 bits.

The signing operation can be particularly expensive. On average, a signature is computed in $t!t^2m^3$ operations. In addition, the public key size is in the order of $tm2^m$.

Concerning the security of the scheme, it is threatened by the usual attacks against code-based cryptography. The most common attack against signature algorithms aims at forging the signature. In code context the latter is faced with the Syndrome Decoding (SD) problem: once the hash of the message is computed, the corresponding error-pattern vector (signature) has to be found. Furthermore, if for the same parity check matrix H and error pattern e, multiples syndromes are tested, then it is enough to solve only one out of many instances. This is referred to as One out of Many Syndrome Decoding (OMSD). This problem can be solved by performing two different attacks:

1. **Information Set Decoding**, it solves instances of SD problem with few solutions in $2^{mt/2}$ operations.
2. **Generalized Birthday Algorithm**, it is suitable for solving SD problems with many solutions.

Over the years, the improvements in the mathematical definition of the attacks has reduced the expected CFS strength. For instance, the GBA work-factor against the proposed CFS parameters, becomes 2^{63} instead of 2^{80}.

Parallel-CFS Because of the GBA potentialities, some years later the same author attempted to improve the CFS security by using parallel CFS signatures [56]. Basically, the parallel-CFS strategy against OMSD relies on the idea of generating i hashes for the same message M using different hashing functions. Therefore, a forger can use the OMSD, but the two guessed error-patterns must be valid for the same M, since the two SD instances are not independent. Furthermore, in order to link the i signatures in a stronger way, the hashes are performed only on M, without appending a counter of the attempt, like in CFS. With respect to the original CFS, the parallel-CFS algorithm is increasing by i times the signature computational effort, the signature size, and the signature verification cost, while maintaining the same key size.

The unique advantage of Parallel-CFS is in terms of security. A successful ISD attack requires the same computational effort of CFS, while a successful GBA is made harder than in the CFS case. It has been proved that the security level of a Parallel-CFS with i equal to 2 or 3 is asymptotically approaching $2^{mt/2}$.

In Table 3.1 we list some interesting parameter settings of the algorithm. It is possible to see that the signature size is compatible with the GNSS constrains. However, an aiding channel would be almost compulsory for accomplishing the distribution of the public key in useful time intervals.

m	t	i	ISD security	orig. CFS security	security against GBA	public key size	sig. cost	sig. size
20	8	3	2^{81}	$2^{66.4}$	$2^{82.5}$	20 MB	$2^{16.9}$	294 bit
18	9	3	$2^{84.5}$	$2^{69.5}$	$2^{83.4}$	5 MB	2^{20}	288 bit
19	9	3	$2^{88.5}$	$2^{72.5}$	$2^{87.7}$	10.7 MB	2^{20}	309 bit

Table 3.1: Parallel-CFS parameters size and corresponding security level.

Digital Signature based on LDGM codes In the State-Of-Art of code-based signatures, a more recent proposal has arisen [57]. It aims at reducing the public key size, and the signing cost of CFS, along with increasing the offered level of security. One of the main limitations of CFS lies in the fact that it uses codes with high rates and low error correction capabilities, in order to reduce the signing time. As a consequence, the scheme is vulnerable to GBA attacks (defeated by using parallel signatures), and the code indistinguishability with respect to a random code is immaterial.

The novel scheme presented in [57] selects only those syndrome vectors with a certain density, and uses the LDGM codes that allow a random-based design. These codes are well-known in the literature for their good error-correcting capability. As is customary for linear codes, they are defined by a generator matrix G, but they can also be constructed in a very simple way, by designing the rows as k random, linearly independent, and n-bit long vectors, with Hamming weight $w \ll n$.

This scheme has the advantage of reducing the public key size with respect to the one of the CFS scheme, at the price of increasing the signature size.

3.2.1.2 Multivariate polynomial cryptography

The Multivariate Public Key Cryptography (MPKC) has been previously tested in order to design cipher algorithms, which have been broken [52]. However, as opposed to codes, these polynomials are showing promising results if used as digital signatures.

The fundamental elements of a MPKC are: a multivariate system F made of quadratic polynomials, which can be easily inverted; and two affine, linear and invertible maps S and T, which have to hide F (or central map). The public key of the system is $P = S \circ F \circ T$ that is difficult to invert, while the private key is made of S, F and T, thus it can invert P. All the variables and coefficients are

taken within a field a finite field $K = GF(q)$. Depending on how F is defined, one can distinguish three different approaches: the BigField (e.g., Matsumoto-Imai that has been the first work, and Hidden Field Equations), the SingleField (e.g., Unbalanced Oil-and-Vinegar and Rainbow schemes), and the MediumField (e.g., $\ell - iC$ schemes). Their security relies on the difficulty of solving systems of multivariate polynomial equations, that is proven to be NP-hard or NP-complete.

The Rainbow signature The Unbalanced Oil-and-Vinegar multivariate structure has been proposed in 1999 [58], then repeated applications of the latter have led to define the Rainbow class of signatures [59]. It is a good candidate as a signature scheme since it allows a fast procedure in both signing and verification.

Several attacks against Rainbow signatures have been proposed. The most famous are the so-called *direct* attacks, and *rank* attacks on which a lot of work has already been done. More recently, *differential* attacks arose, which broke the SFLASH MPKC signature. In [60] their effect has been studied on Rainbow signatures, showing that with the original parameters differential attacks were improving the High-Rank attacks, and even brute-force searches. However, they proposed a novel definition of the Rainbow signature, and an adjustment of its parameters, fixing the issue.

In [61] it is shown that the robustness against the Rainbow-Band-Separation (RBS) attack is strictly related to the cardinality of K, thus they have considered three typical dimensions, i.e., $GF(16)$, $GF(31)$ and $GF(256)$ and proposed varying the size of each parameters over the years depending on the cardinality of K, and including the expected future evolution of key and signature size, based on the Moore's law, as summarized in Table 3.2. It is easy to see that $GF(16)$ is the one that leads to the shortest signature length, while $GF(256)$ requires a small public key for the short term, but its size will grow faster than $GF(31)$ which requires a shorter public key size in the long term.

Depending on the application constraints, the designers can opt for distributing either a longer public key or a longer signature. The signature size lies in the order of some hundreds of bits, while the public key size requires some kB, i.e., [52] reports a signature size of 264 bits, a public key of 842400 bits and a 561352-bit private key for an equivalent 128 bit security level. Therefore, it is possible to think about transmitting the Rainbow signature over the satellite channel in a nearly real-time way, while the longer public key could be shared over a second channel (e.g., the terrestrial data network) in a useful time interval. It must be noted that for a 128-bit security level traditional ECDSA signatures are 512-bit long, thus rainbow signature allows a significant relaxing of the bandwidth requirement.

year	signature (bits)			public key (kB)		
	$GF(16)$	$GF(31)$	$GF(256)$	$GF(16)$	$GF(31)$	$GF(256)$
2010	244	280	344	38.1	30.7	25.7
2020	292	312	424	65.0	44.9	47.5
2030	340	360	520	102.3	72.3	84.0
2040	376	408	592	138.0	99.7	122.6
2050	424	456	680	197.5	138.8	183.3

Table 3.2: Expected signature and public key size growth over years for Rainbow signatures [61].

3.2.2 TESLA-based authentication protocols

Timed Efficient Stream Loss-Tolerant Authentication (TESLA), a broadcast authentication protocol proposed in [62], uses a delayed key disclosure scheme to provide authentication and cryptographic integrity protection of messages on a uni-directional broadcast channel.

The legitimate sender of a message can compute a Message Authentication Code (MAC) at transmission time, being the only entity with prior knowledge of the secret key. Once users have received

a given message with the corresponding MAC, the sender can disclose the used key allowing users to verify the previously received message.

The disclosed key needs to be authenticated, in turn, to ensure both the key and message originated from the legitimate sender. This can be efficiently achieved by verifying the key against a previously authenticated key, using iterations of one-way functions. These are functions that have the following properties:

- *Easy to compute:* for any given input there exist an efficient method to compute the output.
- *Hard to invert:* it is hard to find any input that generates a given output. This is commonly referred to as pre-image resistance, when searching exactly for one particular input value (given $F(m_1)$ shall be difficult to find m_1), and 2nd-preimage resistance when searching for any value different from the actually used input, that produce the same output value (given m_1 shall be difficult to find any $m_2 \neq m_1$ such that $F(m_2) = F(m_1)$).

The sender generates a key-chain of length L starting from a random secret k_L (the initial key) and recursively applying a one-way function $F(\cdot)$, until the last key k_0 (root key) is obtained. The generated key-chain is then used by the sender in the reverse order (Fig. 3.1). Due to the one-way property of the chain, knowledge of key k_i does not give any information on key $k_{i+j}\, \forall j > 0$. The receiver is able to authenticate the received key k_i by applying the one-way function i times to k_i and verifying that the root key k_0 is obtained. The root key must in turn previously be authenticated by other means, such as a digital signature. More efficiently, the verifier can stop applying the one-way function as soon as it reaches a key that he has already verified, that is if $F^{i-j}(k_i) = k_j$ with $j < i$.

TESLA uses the keys from the key-chain for computing MACs. For instance, denoting by M_i the message that the transmitter wants to send at the time i, he uses the key k_i to compute $\mathrm{MAC}_i = \mathbb{S}(M_i, k_i)$ (where $\mathbb{S}$ is the MAC computation algorithm), and sends a packet $P_i = [M_i, \mathrm{MAC}_i, k_{i-d}]$, with $d > 0$. The receiver is not able to verify the received packet $P'_i = \left[M'_i, \mathrm{MAC}'_i, k'_{i-d}\right]$ instantaneously, because he does not know the value of k_i used to compute MAC_i, and has to wait for its disclosure, after d steps. When the user receives the key k'_i he will first check its authenticity and if the result is positive he will compute the MAC for the received data with that key and check if it is equal to the received one: $\mathrm{MAC}_i = \mathbb{S}(M'_i, k'_i) \stackrel{?}{=} \mathrm{MAC}'_i$.

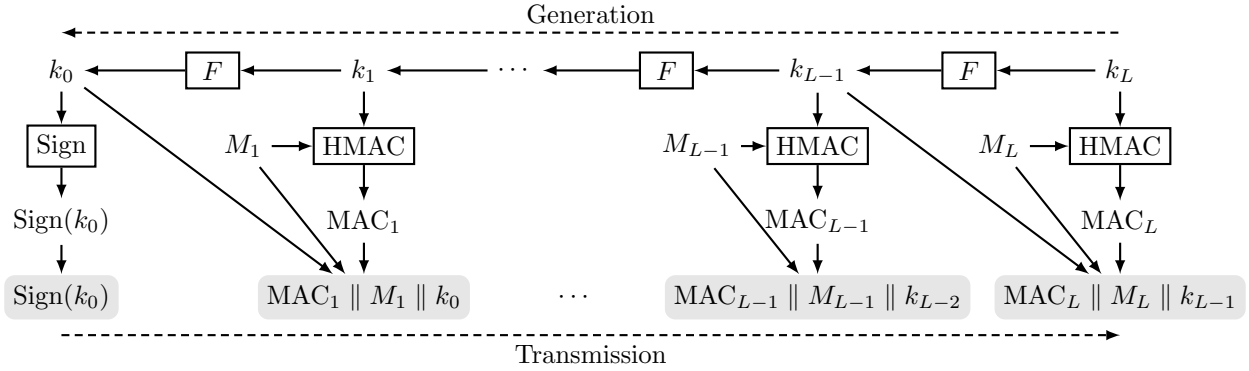


Figure 3.1: TESLA-based authentication.

Various works [48, 49, 50, 51] have described different TESLA based NMA schemes, designed to make use of the External Data Broadcast Service (EDBS) channel for the Galileo Open Service, the reprofiling of the External Region Integrity Service (ERIS) channel once the SoL service was discontinued. The main differences among their works are:

- *Key chain generation:* In [48] it is proposed to build the key chain and the authentication message as:

$$k_i = \text{trunc}(\text{hash}(k_{i+1} \textbf{ xor } w_{\text{pad}} \parallel \text{GST}_i), \ell_{\text{key}}) \tag{3.2}$$

where $w_{\text{pad}} = 1010\ldots10$ is a 128-bit fixed sequence, GST_i is the Galileo System Time (GST), $\ell_{\text{key}} = 128$ represents the length in bits of k_i and $\text{trunc}(x, y)$ denotes the truncation of x to its leftmost y bits.

In [49] it is proposed to build the key chain as:

$$k_i = \text{trunc}(\text{hash}(k_{i+1} \parallel \alpha), \ell_{\text{key}}) \tag{3.3}$$

where α is a binary sequence unique for every key chain that is disclosed at the beginning of the key chain, and $\ell_{\text{key}} = 80$ bits. Such construction was modified in [50] with the inclusion of the GST in the computation:

$$k_i = \text{trunc}(\text{hash}(k_{i+1} \parallel \text{GST}_i \parallel \alpha), \ell_{\text{key}}). \tag{3.4}$$

- *Number of keys used:* In [48] it is proposed that all the SVs use the same key and that a key is revealed every 30 seconds, allowing the verification of the corresponding MAC. In [49, 50, 63] instead it is proposed that every SV uses a different key, but all taken from the same key chain. The concept is to assign a new key to each SV so that even in the same authentication round each SV could broadcast a different key. This was proposed in order to avoid the possibility that an attacker could take advantage of the different propagation delays that could allow to derive the secret key from high elevation SVs and replay it to the low elevation SVs. To prevent this, 40 keys from the same key chain are used in each round.

3.3 Analysis of state-of-the-art techniques

3.3.1 Discussion on TESLA-based authentication protocols

The main drawbacks and limitations of the current proposed approaches and variations are:

- *Time synchronization requirement:* TESLA is standardized for use in multicast applications [64]. One of the requirements for the security of the scheme is a (loose) time synchronization between the receiver and the sender. Indeed, after the disclosure of the key by the system, this can be used to compute the MAC of any arbitrary message. For this reason, the receiver shall check that the MAC is received before the disclosure of the corresponding key by the system, but in order to perform this verification some synchronization is required. In the multicast over bi-directional link context, such a requirement is easily fulfilled with traditional mechanisms. However, when TESLA is applied to GNSS, that itself used as a time reference, this requirement is far less obvious, especially for autonomous receivers at start-up after a long off period.

- *Vulnerabilities to pre-computatation attacks:* in the case of [49] where the padding is fixed for the whole chain length. This vulnerability will be discussed in Section 3.3.2. This issue was acknowledged and fixed in [50].

- *Security of the key chain:* the original TESLA scheme is considered secure, and various security proofs were presented in literature [65, 66, 67]. A basic assumption to this security proofs is the full entropy of each key and the time synchronization between transmitter and the receiver. However, a common construction in the three proposals listed above is the truncation of the hash output in the key chain in order to reduce the bandwidth requirements, and the subsequent padding when used in the next iteration. The impact of this modification on the security of

the scheme will be evaluated in Section 3.3.3, showing that the iterative operations of padding-hashing-truncation leads to a reduction in the entropy of the chain itself. In order to reduce the impact of this sub-optimal construction, a careful choice of the system parameters is needed, preferring longer keys and fewer steps in the key chain.

- *Security of the MAC:* the MAC construction has been tuned as well, truncating it down to just 10 bits. However, any reduction on bandwidth comes at the cost of a reduction of the security of the scheme. In TESLA, the authentication of data is provided by the MACs, and while a single few bits long MAC is fine for providing an ephemeral authentication that last for few seconds, it is not adequate for providing an authentication that last for several hours. This means that, in order to achieve the desired security level, a receiver might be required to verify multiple MACs.

3.3.2 Security evaluation of ideal one-way chains

Two proposals for the construction of the key chain were introduced. One foresees the use of a time dependent parameter (see Eq (3.2), (3.4)) in order to prevent the attacker from using precomputed pieces of the key chain to perform attacks in later instants. In (3.3) instead this time varying padding is not present.

In the following, a brute force attack will be formalized and its success probability will be computed.

We assume that the attacker has a finite memory that can fit B keys, he is able to compute R_H hashes per second, he wants to perform an attack that must last for D keys and he is willing to wait up to T seconds to perform the attack. The system is using a key of k bits so the cardinality of the key space is $N = 2^k$, and the keys are disclosed by the system at the rate R_K.

The attacker has a finite computational power and the computation of a chain of D keys takes

$$T_{KC} = \frac{D}{R_H} \tag{3.5}$$

and his memory can fit

$$N_{KC} = \min\left(\left\lfloor \frac{B}{D} \right\rfloor, \left\lfloor \frac{T}{T_{KC}} \right\rfloor\right) \tag{3.6}$$

different chains. A smarter solution, that allows to increase the number of buffered chains, is to store only the first and the last key of the each chain. If a buffered final key is reached the attacker knows that he can compute the next piece of the key chain starting from the corresponding initial key. This leads to a number of key chains buffered equal to

$$N_{KC} = \min\left(\left\lfloor \frac{B}{2} \right\rfloor, \left\lfloor \frac{T}{T_{KC}} \right\rfloor\right) \tag{3.7}$$

After computing a key chain, in the remaining time the system will disclose $R_K(T - T_{KC})$ keys and thus the probability that one of the disclosed keys is equal to the first of the computed key chain is:

$$P_S = \frac{R_K(T - T_{KC})}{N} \tag{3.8}$$

The number of keys disclosed after the computation of the second key chain is $R_K(T - 2T_{KC})$, after the third is $R_K(T - 3T_{KC})$ and so on, thus we can write the success probability as the union bound

$$\begin{aligned} P_S &= \frac{R_K(T - T_{KC})}{N} + \frac{R_K(T - 2T_{KC})}{N} + \cdots + \frac{R_K(T - N_{KC}T_{KC})}{N} \\ &= \frac{R_K}{N}\sum_{i=1}^{N_{KC}}(T - iT_{KC}) = \frac{R_K N_{KC}}{N}\left[T - T_{KC}\frac{(N_{KC}+1)}{2}\right] \end{aligned} \tag{3.9}$$

Using Intel® optimized assembly code [68] on an Intel® Core™i5-5257U CPU we obtained a hashing rate of $R_H = 3.5 \cdot 10^6$ Secure Hash Algorithm (SHA)-256 hash/s. On the other hand SHA-256 is also used by Bitcoin and specialized powerful and efficient hardware can be found in the market for Bitcoin mining that use Application Specific Integrated Circuit (ASIC). As an example modern hardware allow achieving hashing rates in the order to 10^{13} hash per seconds and costs around one thousand US dollars [69]. Bitcoin user organized themselves in pools in order to cooperate. These pools aggregate the computational power of many users and can reach a cumulative hashing rate in the order of $490 \cdot 10^{15}$ hash per second [70]. [71] reports the historic aggregate hashing rate of the Bitcoin network.

Suppose that the attacker wants to perform an attack that lasts for 60 seconds and so he needs to precompute key chains of $D = 240$ keys each if the system discloses $R_K = 40/10$ keys per seconds. Fig. 3.2a shows the success probability as a function of the available memory and of the elapsed time from the start of the attack, against the time invariant construction (3.3) with key length of 82 bits.

It is important to note that the success probability is limited by the memory size rather than by the computational power of the attacker.

In the time variant construction, (3.2)-(3.4), instead, the attacker can only fix the time interval in which he wants to perform the attack and attempt to compute all the possible chains that his memory can fit, that are N_{KC} (3.7) so the success probability of an attack performed against a single key release is,

$$P_S = \frac{N_{KC}}{N} \tag{3.10}$$

because every computed chain has only one chance to be valid. It is possible to modify the attack model as follows: the attacker is able to compute R_H hash per second, thus he can compute

$$N_{KC} = \frac{R_H T_d}{D} \tag{3.11}$$

pieces of key chain between two key disclosures. At every release instant of keys if the attacker does not find a collision, he can recompute N_{KC} new key chains before the next key release. The success probability of the attack is

$$P_S(t) = \frac{N_{KC}}{N} \cdot \left\lfloor \frac{t}{T_d} \right\rfloor, \quad t \leq \text{duration of key chain} \tag{3.12}$$

and the attacker needs to store $2 \cdot N_{KC}$ values in his memory. In this case the success probability is rather limited by the attacker's computational power than by the memory size.

Fig. 3.2b shows the success probability using the same attack parameters as before and 128-bit keys disclosed with a rate $R_K = 1/30$ key per second.

The above discussed attack is somehow similar to the time-memory trade-off introduced in [73]. In that case the goal of the attack was to break Data Encryption Standard (DES) but it can be used to invert one-way functions. A fundamental difference between the two attacks is that in [73] the event of finding a second preimage (an input value different from the authentic one that leads to the same output value) represents *false alarm*, and are a missed shot for the attacker, while here this becomes a valid output of the attack and contribute to the success probability.

3.3.3 Security evaluation of one-way chains generation algorithm

In order to evaluate both the performance and the security of the key generation algorithm, this was implemented and some experimentation was performed. An important result is that computing the hash of all the possible 256 inputs of 8 bits, only 158 different outputs were reached. Clearly this can not be representative of the implementation using much longer keys (i.e., 80-128 bits) and it can be seen as a particular realization of the key generation algorithm in terms of padding used.

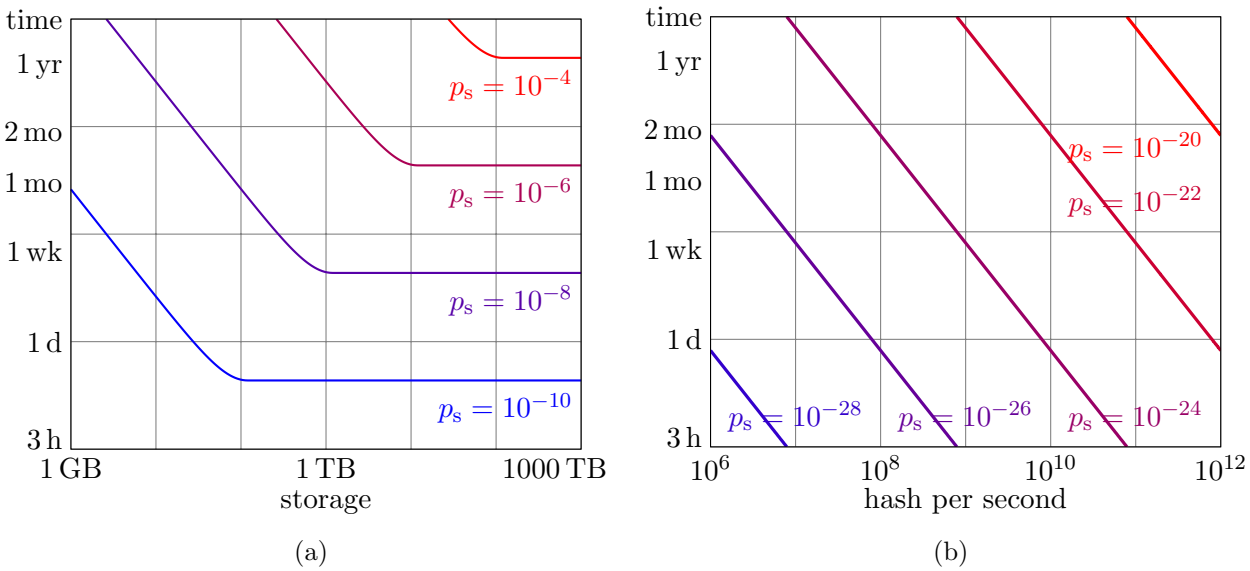


Figure 3.2: Success probability of a brute force attack against the key chain with ideal hashing function. The key generation algorithm is: (a) time invariant [72] and (b) time variant [48, 50].

Many investigations were performed, showing that the result is fundamentally not influenced by the used hashing function (MD5 and SHA-1 perform only marginally worse than SHA-256) and that even randomly selecting the output bits in the output words instead of the Most Significant Bits (MSBs) does not solve the issue.

Moreover, at every performed iteration the output space keeps reducing. Indeed, by applying again the SHA-256 hashing function to the 158 possible output obtained at the first iteration only 113 distinct values are obtained.

A pictorial representation of this problem is given in Fig. 3.3. In this example only $N = 24$ different values are taken into account and the hashing function is replaced by a random function (this model will be motivated in the next Section). Applying the random function to all the possible key k_L the set of the possible key k_{L-1} is obtained. It is easy to see that many of the starting values collide, and that this repeats at every iteration, even if the random function is updated at every step emulating a time varying key generation function.

Fig. 3.4 gives a pictorial representation of the padding-truncation construction. Assuming that key generation starts from a completely random value with full entropy, the combination of the padding and hash function is applied to it. The hash function is designed and generally used as a compression function that, starting from an arbitrary length input produces a shorter, fixed length output uniformly distributed in the output space. Instead, in the construction proposed in [48, 49, 50], the combination of padding and hash function is used as an expansion function, that produces an output longer than the input. No fresh entropy is included in the computation, because if some unpredictable information is used in the computation, this information shall be transmitted and authenticated to the receiver in order to replicate the computation, leading to a higher bandwidth requirement. It must be noted that the inclusion of time-varying information such as the Galileo System Time (GST) does not increase the entropy because this information is predictable. Due to the lack of new entropy, the padding-hash operation works as an expansion function that spreads the entropy of the input values in the output space. The following operation in the proposed construction is a truncation. This can be seen as a deterministic compression function. Intuitively, this compression function, in general, does not preserve all the entropy in the hash function output, and produce a key with a smaller entropy than the previous one. Then, the process is iterated, leading to a progressive loss of entropy along the key

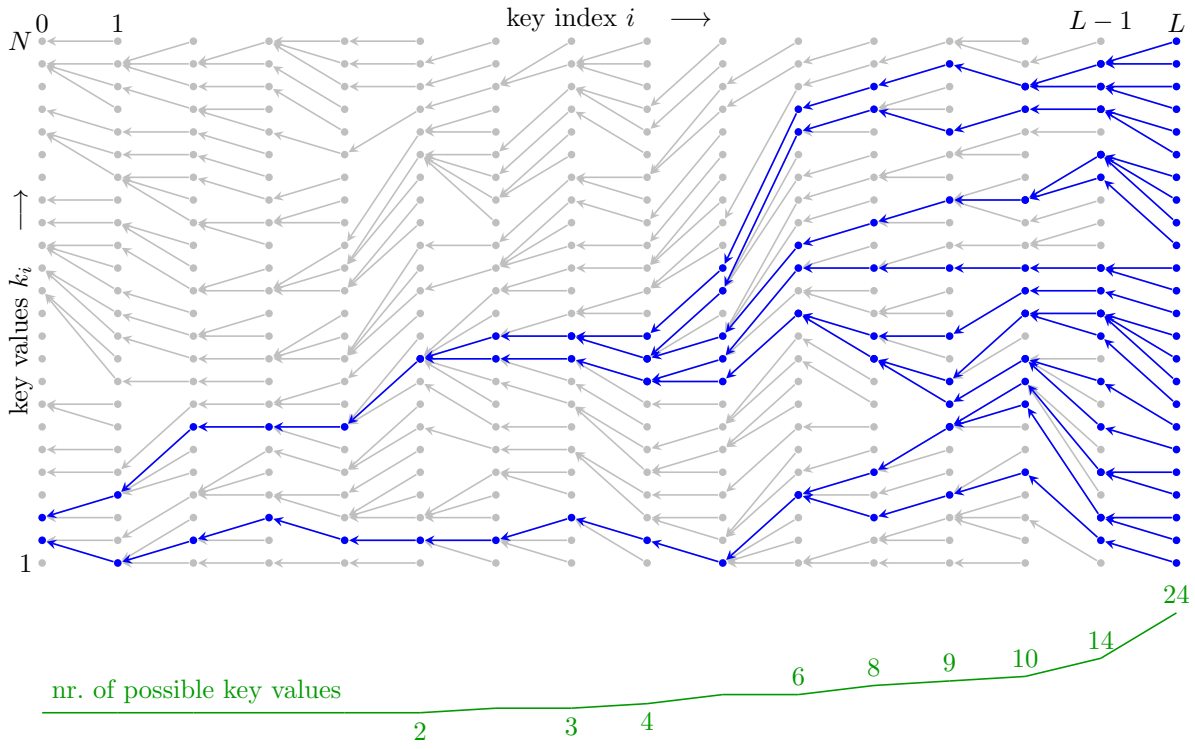


Figure 3.3: Output of the key generation function.

chain.

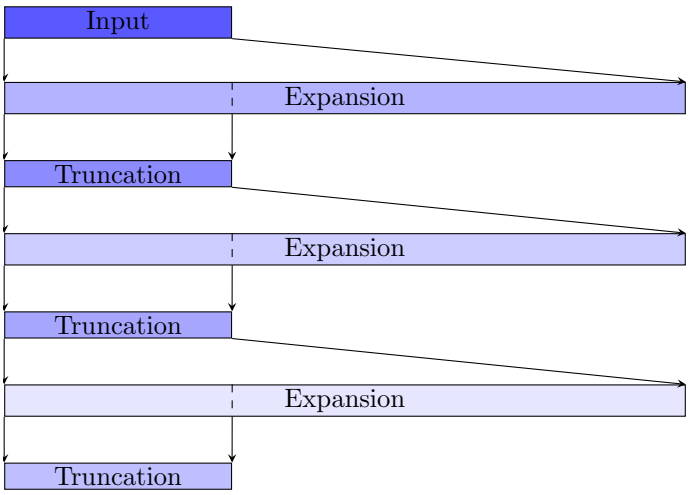


Figure 3.4: Pictorial representation of the padding-truncation construction of the TESLA key chain.

3.3.3.1 Probabilistic model of the one-way key chain

The (possibly time varying) n-bit hash function in the TESLA key chain is built as illustrated in Fig. 3.5a, by cascading

1. a possibly time-varying, yet deterministic padding to increase the length from n to $m' > n$ bits
2. a secure cryptographic hash function with m'-bit inputs and m-bit outputs, taken from a well

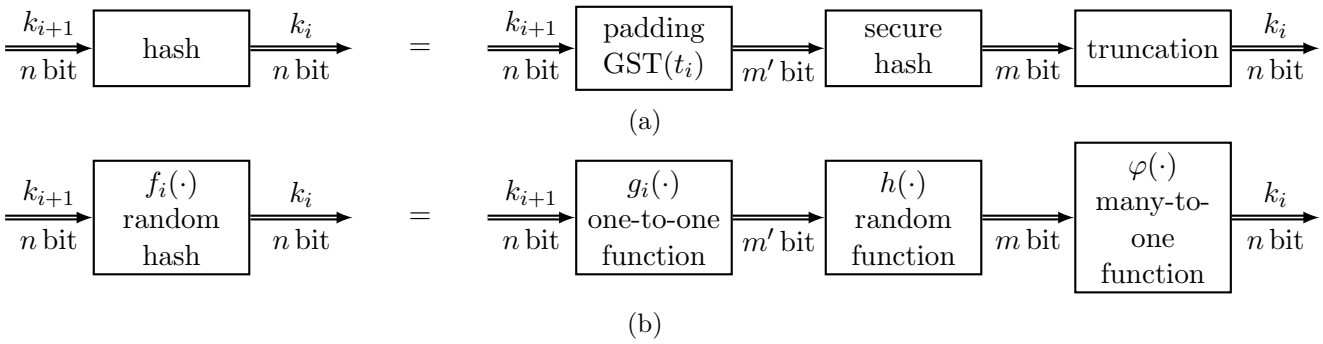


Figure 3.5: (a) construction of a $n \to n$ bit hash function for the TESLA key chain starting from a $m' \to m$ secure hash function with $m, m' > n$, as proposed in [50]; (b) probabilistic model for its security analysis.

established cryptographic library

3. a truncation of the secure hash output to n bits

A security analysis of the hash chain would require to evaluate the effectiveness of the hash function, in terms of uniformity of its output, by exhaustively generating all 2^n input/output pairs, for each possible value of the time varying parameter GST(t). Clearly this is not feasible even for moderate values of n, so for the sake of tractability, we replace the explicit construction with a probabilistic model.

In particular, we model the $m' \to m$ bit secure hash function as a one-to-one function, thus assuming $m' \leq m$. This represents an idealization of the actual hash function, in that we neglect the possibility of having collisions, since we are interested in the additional collisions that may be introduced by concatenation with padding and truncation. A more pessimistic choice would be to model the hash function as a random oracle. On the other hand, since the m-bit secure hash function was not designed to have a uniform output when truncated to n bits with arbitrarily padded n-bit inputs, we consider that in this respect it is as good as any other m'-bit to m-bit, one-to-one, map. Therefore, we model it as a random function h, uniformly drawn from the set of all one-to-one functions with the same input and output sets. Correspondingly, the time-varying padding is modeled as a deterministic one-to-one map, while the truncation is a many-to-one map. The resulting model is represented in Fig. 3.5b.

Under the simplifying model assumptions that

- each $g_i : \mathcal{N} \to \mathcal{M}'$, $i = 0, \ldots, L-1$ is a one-to-one function with $|\mathcal{N}| = N = 2^n$ and $|\mathcal{M}'| = M' = 2^{m'}$
- $h : \mathcal{M}' \to \mathcal{M}$ is a random one-to-one function (uniformly chosen), with $|\mathcal{M}| = M = 2^m$
- $\varphi : \mathcal{M} \to \mathcal{N}$ is a balanced many-to-one function with $\left|\varphi^{-1}(\{b\})\right| = M/N = 2^{m-n} \gg 1$, $\forall b$

each $f_i : \mathcal{N} \to \mathcal{N}$, $f_i = \varphi \circ h \circ g_i$ can be modeled as an independent random oracle, i.e., $\{f_i(a)\}$, $a \in \mathcal{N}, i = 0, \ldots, L-1$ are NL independent random variables, each uniformly distributed in $\mathcal{N}$. This holds even more if h is itself modeled as a random oracle.

For each $b \in \mathcal{N}, i = 0, \ldots, L-1$, let us denote by $K_i(b) = \left|f_i^{-1}(\{b\})\right|$ the cardinality of the preimage of b under f_i, and by $J_{i,j}(b) = |[\boldsymbol{f}_i^j]^{-1}(\{b\})|$, that of the preimage of b under $\boldsymbol{f}_i^j = f_i \circ f_{i+1} \circ \ldots \circ f_{j-1}$. By natural extension, $\boldsymbol{f}_i^i$ is the identity map on $\mathcal{N}$ and $J_{i,i}(b) = 1$, $\forall b$.

Modeling f_i as a random oracle, the random vector $\boldsymbol{K}_i = [K_i(1), \ldots, K_i(N)]$ has the $(N, \frac{1}{N})$ multinomial distribution

$$p_{\boldsymbol{K}_i}(\boldsymbol{c}) = \begin{cases} 0 & , \ \sum_{b\in\mathcal{N}} c_b \neq N \\ \dfrac{N!}{\prod_{b\in\mathcal{N}}(c_b!)}\dfrac{1}{N^N} & , \ \sum_{b\in\mathcal{N}} c_b = N \end{cases} \tag{3.13}$$

and the marginal distribution of each $K_i(b)$ is a $(N, \frac{1}{N})$ binomial

$$p_{K_i}(c) = \binom{N}{c}\frac{1}{N^c}\left(1-\frac{1}{N}\right)^{N-c} \tag{3.14}$$

For $N \gg 1$ the $(N, \frac{1}{N})$ binomial distribution can be approximated by a Poisson distribution with unit mean

$$p_{K_i}(c) \simeq \frac{1}{c!e} \tag{3.15}$$

On the other hand, the random variable $J_{i,L}(b)$ can be written through a backward recursion on i as

$$J_{i,L}(b) = \sum_{a\in f_i^{-1}(\{b\})} J_{i+1,L}(a) \tag{3.16}$$

If the f_i are also modeled as independent, and we take into account the fact that $J_{i,L}(b), K_i(b) \ll N$ with high probability, the $K_i(b)$ random variables $J_{i+1,L}(a)$ in the sum above can be viewed as independent and identically distributed (i.i.d.) and independent of $K_i(b)$, so that $J_{i,L}(b)$ has a *compound Poisson* distribution, with mean and power given by

$$\mathrm{E}\left[J_{i,L}\right] = \mathrm{E}\left[K_i\right]\mathrm{E}\left[J_{i+1,L}\right] = \mathrm{E}\left[J_{i+1,L}\right] \tag{3.17}$$

$$\mathrm{E}\left[J_{i,L}^2\right] = \mathrm{E}\left[K_i\right]\left(\mathrm{E}\left[J_{i+1,L}^2\right] + \mathrm{E}\left[J_{i,L}\right]\right) = \mathrm{E}\left[J_{i+1,L}^2\right] + 1 \tag{3.18}$$

Starting from $J_{L,L}(b) = 1$, and by applying the recursive relations above, we obtain

$$\mathrm{E}\left[J_{i,L}\right] = 1 \quad , \quad \mathrm{E}\left[J_{i,L}^2\right] = L - i + 1 \tag{3.19}$$

Conditioned on the actual realization of $\boldsymbol{f}_i^L$, the distribution of k_i, when k_L is uniformly distributed in $\mathcal{N}$ is given by

$$\mathrm{P}\left[k_i = b|\boldsymbol{f}_i^L\right] = \frac{J_{i,L}(b)}{N} \tag{3.20}$$

The corresponding collision probability at step i, that is the probability that for k_L, k'_L uniformly and independently chosen in $\mathcal{N}$, $k_i = \boldsymbol{f}_i^L(k_L)$ and $k'_i = \boldsymbol{f}_i^L(k'_L)$ coincide, is given by

$$p_{\mathrm{c}} = \mathrm{P}\left[k_i = k'_i|\boldsymbol{f}_i^L\right] = \frac{1}{N^2}\sum_{b\in\mathcal{N}} J_{i,L}^2(b) \tag{3.21}$$

and is itself a random variable. We then obtain its expected value

$$\mathrm{E}\left[p_{\mathrm{c}}\right] = \frac{1}{N}\mathrm{E}\left[J_{i,L}^2\right] = \frac{L-i+1}{N} \tag{3.22}$$

and observe that it increases linearly by each step of the chain, as long as $L \ll N$. Compare this to the ideal case where $p_{\mathrm{c}} = 1/N$.

Another measure of how the key chain deviates from an ideal hashing distribution can be obtained by looking at the number of possible values for k_i, that is the cardinality of the image of $\mathcal{N}$ under $\boldsymbol{f}_i^L$,

$$N_i = \left|\boldsymbol{f}_i^L(\mathcal{N})\right| = \left|\{b \in \mathcal{N} \ : \ J_{i,L}(b) > 0\}\right| \tag{3.23}$$

Under the above model, N_i is a random variable itself

$$N_i = \sum_{b \in \mathcal{N}} \chi_{\{J_{i,L}(b)>0\}} \tag{3.24}$$

where χ_A denotes the indicator variable of event A. The mean of N_i is then

$$\mathrm{E}\left[N_i\right] = \sum_{b \in \mathcal{N}} \mathrm{P}\left[J_{i,L}(b) > 0\right] = N(1 - p_{J_{i,L}}(0)) \tag{3.25}$$

where one can again use the properties of the compound Poisson distribution to derive the recursive relationship

$$p_{J_{i,L}}(0) = e^{p_{J_{i+1,L}}(0)-1} \tag{3.26}$$

with the initial value $p_{J_{L,L}}(0) = 0$. From (3.25), and by Jensen inequality, one can also derive an upper bound on the equivocation (conditional entropy) of each key k_i given the hash functions

$$H(k_i|\boldsymbol{f}_i^L) \leq \mathrm{E}\left[\log_2 N_i\right] \leq \log_2 \mathrm{E}\left[N_i\right] \tag{3.27}$$

A numerical evaluation of the collision probability (3.22) and the entropy bound (3.27) is shown in Fig. 3.6 for the TESLA parameters proposed in [50] where $n = 80\,\mathrm{bit}$, $m = 256\,\mathrm{bit}$ and key k_i is released at time $t_i = t_0 + iT_k$, with $T_k = 0.25\,\mathrm{s}$ the key release period (40 keys every 10 seconds).

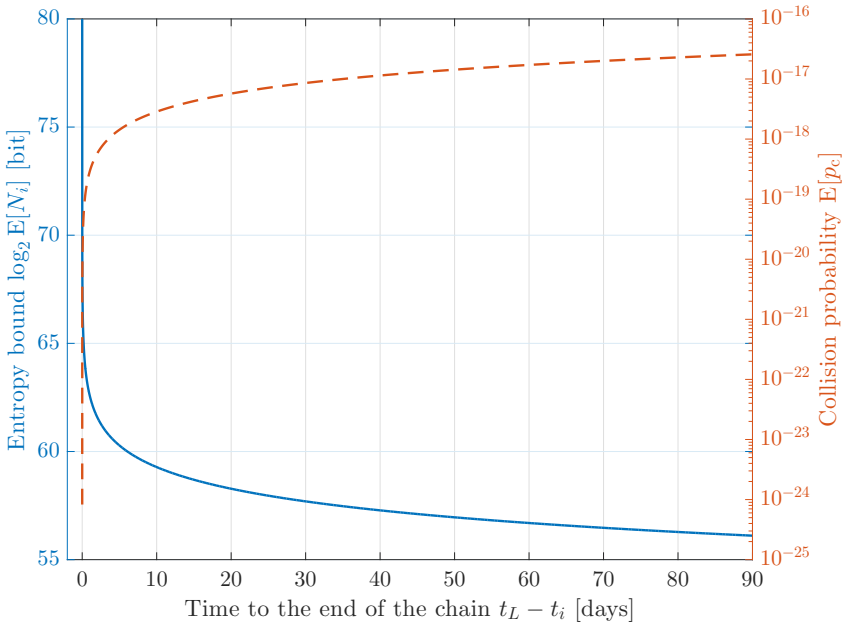


Figure 3.6: Upper bound on the entropy (solid line) and expected collision probability (dashed lines) after each iteration for 80-bit key chain.

3.3.3.2 Numerical validation of the model

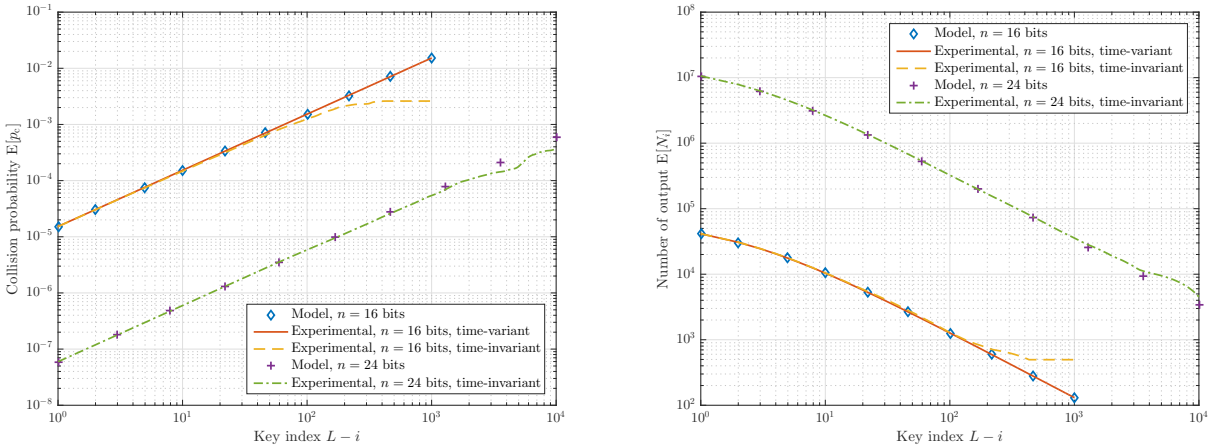
This Section compares results from Section 3.3.3.1 with those obtained by a numerical implementation.

Due to computational burden only shorter key lengths (i.e., $n = 16$ and $n = 24$ bits) were evaluated, using a standard SHA-256 as the internal secure hash function in Fig. 3.5a. The implementation iteratively computes all the keys k_i, $i = L-1, \ldots, 0$ starting from each possible value of the input k_L.

For example, for 24-bit key chain, the computation starts from all the 2^{24} possible inputs, truncating the output of the hash function to 24 bits for each iteration.

In Fig. 3.7a, a comparison of the predicted versus experimental collision probability is provided, taking into account a truncation to $n = 16$ bits both with time-variant and time-invariant padding, and to $n = 24$ bits with time-invariant padding only for computational limitations. The figure illustrates a perfect correspondence between the predicted and experimental values for the 16-bits time-variant key generation algorithm case. At the beginning of the key chain a good agreement can be observed also for the versions with time-invariant padding, while a difference can be observed after 400 steps for 16-bit keys, and after 2000 steps for 24-bit keys. This discrepancy is due to the fact that in this case the hash function remains the same at each iteration and it is poorly modeled by the assumption of independence between the f_i. In fact, after a certain number of iterations the chain reaches a limit cycle, the image $\boldsymbol{f}_i^L(\mathcal{N})$ remains identical, and the output distribution with uniform input becomes stationary. Clearly, the longer the key, the more iterations are needed to reach this situation, thus the model remains valid for more iterations also in the presence of time-invariant padding.

Fig. 3.7b reports the comparison between the experimental and the theoretical numbers of distinct outputs N_i after every iteration, for 24-bit time-invariant and 16-bit time-variant/invariant key chain, respectively. Once again, the average results match the predicted ones when a time-variant function is used, and the model is a good approximation for the initial iterations of the time-invariant key generation algorithm.



(a) (b)

Figure 3.7: Comparison between predicted and experimental (a) collision probability; (b) number of possible output after each iteration.

3.3.3.3 Attack model

We consider an attack that aims at spoofing the navigation message for a time duration $T_A = \ell T_k$, beginning at t_i, i.e., at having the victim accept ℓ consecutive keys and the corresponding forged messages and MACs as authentic.

To this purpose, the attacker tries to find $\hat{k}_{i+1}, \ldots, \hat{k}_{i+\ell}$ such that $f_i(\hat{k}_{i+1}) = k_i$, and $f_{i+1}(\hat{k}_{i+2}) = \hat{k}_{i+1}$, …, $f_{i+\ell-1}(\hat{k}_{i+\ell}) = \hat{k}_{i+\ell-1}$.

He can pick N_A random, and independent values within $\mathcal{N}$ as guesses for $k_{i+\ell}$,

$$\hat{k}_{i+\ell}^j \sim \mathcal{U}(\mathcal{N}) \quad , \quad j = 1, ..., N_A \tag{3.28}$$

and recursively compute the chain of ℓ consecutive hashes up to

$$\hat{k}_i^j = \boldsymbol{f}_i^{i+\ell}(\hat{k}_{i+\ell}^j) \quad , \quad j = 1, ..., N_A \tag{3.29}$$

For each guess, the attacker checks whether $\hat{k}_i^j$ coincides with the actual disclosed k_i. If so, he can use the values $(\hat{k}_{i+1}^j, \ldots, \hat{k}_{i+\ell}^j)$ as TESLA keys for his spoofed messages and have the victim accept them as authentic, otherwise he goes on with the search, up to the N_A-th attempt.

His choice of N_A is constrained by the computational power that is available to him (in terms of hashing rate R_h, i.e., the maximum number of hashes he can compute per unit time, and memory required to buffer the precomputed values) and the amount of time T that he is willing to devote to the attack computation.

Note that the key values $\hat{k}_{i+1}, \ldots, \hat{k}_{i+\ell}$ computed by the attacker may be different from the corresponding ones computed by the system, and the attack would still be successful, as long as $\hat{k}_i$ is equal to k_i. In fact the attacker aim is not to find the exact key sequence released by the system, but any sequence that passes the verification.

The attack is represented in Fig. 3.8. Assuming that the system uses the legitimate key chain colored in green and that the attacker aims at attacking the highlighted portion of the key chain between the key k_i and the key $k_{i+\ell}$, he can pick a random staring point $\hat{k}_{i+\ell}$ and apply ℓ times the one-way function. For an ideal key chain generation function we expect that there is only one valid guess, that is the key used by the system. It is evident from the figure that there are much more valid starting points. For the particular realization of the example, 17 values out of the 24 possible values lead to the correct key chain after 7 iterations. If the attacked interval was shorter the number of valid key would be smaller: for a single key guessing, obviously, only one of the possible 24 keys would be valid, for a sequence of 2 keys there would be 2 valid keys; for a series of 4 keys there would be 4 possible keys; and for 5 consecutive keys the number of valid key would grow to 8. This example shows that the success probability increases almost linearly with the number of iterations of the key generation function attacked. In the following the success probability will be derived using the probabilistic model.

The success event for this attack can be written as

$$\mathbb{S}(i, \ell, N_A) = \bigcup_{j=1}^{N_A} S_j(i, \ell) \tag{3.30}$$

where the success event for a single guess i is

$$S_j(i, \ell) = \{\hat{k}_i^j = k_i\} \ . \tag{3.31}$$

Similarly to the derivation of the collision probability in Section 3.3.3.1, we observe that, conditioned on the actual realization of $\boldsymbol{f}_i^{i+\ell}$, the $\hat{k}_i^j$ are i.i.d. and independent of k_i with probability mass distribution (pmd)

$$\mathrm{P}\left[\hat{k}_i^j = b | \boldsymbol{f}_i^{i+\ell}\right] = \frac{J_{i,i+\ell}(b)}{N} \tag{3.32}$$

Then we can evaluate the conditioned single attempt success probability

$$\mathrm{P}\left[S_j(i, \ell) | \boldsymbol{f}_i^L\right] = \sum_{a \in \mathcal{N}} \mathrm{P}\left[k_i = a | \boldsymbol{f}_i^L\right] \mathrm{P}\left[\hat{k}_i^j = a | \boldsymbol{f}_i^{i+\ell}\right] \tag{3.33}$$

$$= \sum_{a \in \mathcal{N}} \frac{1}{N^2} J_{i,L}(a) J_{i,i+\ell}(a) \tag{3.34}$$

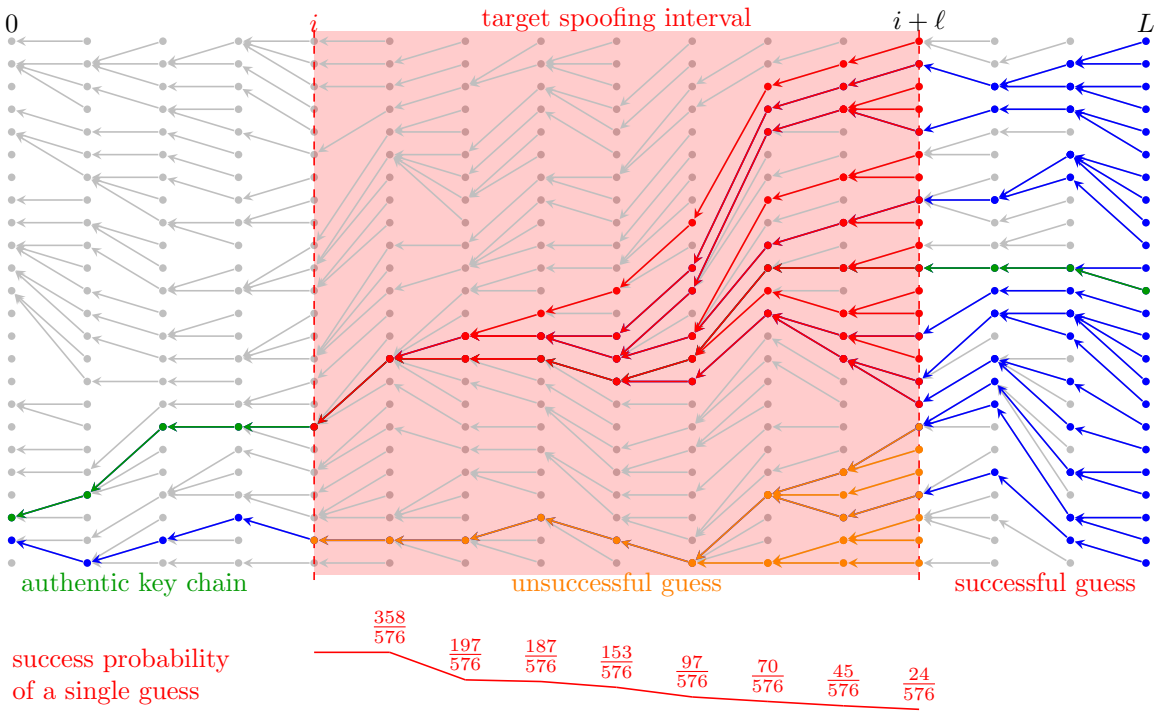


Figure 3.8: Pictorial representation of the attack against TESLA key chain.

On the other hand, one can recursively write the product as

$$J_{i,L}(b)J_{i,i+\ell}(b) = \left(\sum_a J_{i+1,L}(a)\right)\left(\sum_{a'} J_{i+1,i+\ell}(a')\right) \tag{3.35}$$

$$= \sum_a J_{i+1,L}(a)J_{i+1,i+\ell}(a) + \sum_{a,a'\neq a} J_{i+1,L}(a)J_{i+1,i+\ell}(a') \tag{3.36}$$

where all the sum indices a, a' run over $f_i^{-1}(\{b\})$. The first sum in (3.36) yields again a compound Poisson variable, whereas in each term of the second sum, $J_{i+1,L}(a)$ and $J_{i+1,i+\ell}(a')$ are independent, so we obtain the recursion

$$\begin{aligned}\mathrm{E}\left[J_{i,L}(b)J_{i,i+\ell}(b)\right] &= \mathrm{E}\left[K_i\right]\mathrm{E}\left[J_{i+1,L}(a)J_{i+1,i+\ell}(a)\right] \\ &\quad + \mathrm{E}\left[K_i(K_i-1)\right]\mathrm{E}\left[J_{i+1,L}(a)\right]\mathrm{E}\left[J_{i+1,i+\ell}(a)\right] &(3.37)\\ &= \mathrm{E}\left[J_{i+1,L}(a)J_{i+1,i+\ell}(a)\right] + 1 &(3.38)\end{aligned}$$

which, together with the starting point $J_{i+\ell,i+\ell}(b) = 1$, $\forall b$, yields the result

$$\mathrm{E}\left[J_{i,L}(b)J_{i,i+\ell}(b)\right] = \ell + 1 \tag{3.39}$$

We then obtain the average single attempt success probability as

$$\mathrm{P}\left[S_j(i,\ell)\right] = \sum_{a\in\mathcal{N}} \frac{1}{N^2}\mathrm{E}\left[J_{i,L}(a)J_{i,i+\ell}(a)\right] = \frac{\ell+1}{N}\ . \tag{3.40}$$

Similarly, we can derive the joint success probability for two distinct guesses j and j' on the same key chain section $k_i, \ldots, k_{i+\ell}$, that is $\mathrm{P}\left[S_j(i,\ell) \cap S_{j'}(i,\ell)\right]$. We start by conditioning on the realization of

the sequence of hash functions $\boldsymbol{f}_i^L$

$$\mathrm{P}\left[S_j(i,\ell)\cap S_{j'}(i,\ell)|\boldsymbol{f}_i^L\right] = \mathrm{P}\left[\hat{k}_i^j = k_i, \hat{k}_i^{j'} = k_i|\boldsymbol{f}_i^L\right] \tag{3.41}$$

$$= \sum_{a\in\mathcal{N}} \mathrm{P}\left[k_i = a, \hat{k}_i^j = a, \hat{k}_i^{j'} = a|\boldsymbol{f}_i^L\right] \tag{3.42}$$

$$= \sum_{a\in\mathcal{N}} \mathrm{P}\left[\boldsymbol{f}_i^L(k_L) = a, \boldsymbol{f}_i^{i+\ell}(\hat{k}_{i+\ell}^j) = a, \boldsymbol{f}_i^{i+\ell}(\hat{k}_{i+\ell}^{j'}) = a|\boldsymbol{f}_i^L\right] \tag{3.43}$$

and by leveraging the fact that $k_L, \hat{k}_{i+\ell}^j, \hat{k}_{i+\ell}^{j'}$ are chosen independently and uniformly over $\mathcal{N}$ we get

$$\mathrm{P}\left[S_j(i,\ell)\cap S_{j'}(i,\ell)|\boldsymbol{f}_i^L\right] = \sum_{a\in\mathcal{N}} \mathrm{P}\left[\boldsymbol{f}_i^L(k_L) = a|\boldsymbol{f}_i^L\right]\cdot \mathrm{P}\left[\boldsymbol{f}_i^{i+\ell}(\hat{k}_{i+\ell}^j) = a|\boldsymbol{f}_i^{i+\ell}\right]\cdot \mathrm{P}\left[\boldsymbol{f}_i^{i+\ell}(\hat{k}_{i+\ell}^{j'}) = a|\boldsymbol{f}_i^{i+\ell}\right]$$

$$= \sum_{a\in\mathcal{N}} \frac{J_{i,L}(a)}{N}\frac{J_{i,i+\ell}(a)}{N}\frac{J_{i,i+\ell}(a)}{N} \tag{3.44}$$

$$= \frac{1}{N^3}\sum_{a\in\mathcal{N}} J_{i,L}(a)J_{i,i+\ell}^2(a) \tag{3.45}$$

Then, the overall probability can be obtained by averaging the above result over the distribution of $\boldsymbol{f}_i^L$. To this purpose we can recursively write the product as

$$J_{i,L}(a)J_{i,i+\ell}^2(a) = \left(\sum_b J_{i+1,L}(b)\right)\left(\sum_{b'} J_{i+1,i+\ell}(b')\right)\left(\sum_{b''} J_{i+1,i+\ell}(b'')\right) \tag{3.46}$$

$$\begin{aligned}
&= \sum_b J_{i+1,L}(b)J_{i+1,i+\ell}^2(b) + \sum_b\sum_{b'\neq b} J_{i+1,L}(b)J_{i+1,i+\ell}^2(b') \\
&+ \sum_b\sum_{b'\neq b} J_{i+1,L}(b)J_{i+1,i+\ell}(b')J_{i+1,i+\ell}(b) + \sum_b\sum_{b'\neq b} J_{i+1,L}(b)J_{i+1,i+\ell}(b)J_{i+1,i+\ell}(b') \\
&+ \sum_b\sum_{b'\neq b}\sum_{b''\neq b,b'} J_{i+1,L}(b)J_{i+1,i+\ell}(b')J_{i+1,i+\ell}(b'')
\end{aligned} \tag{3.47}$$

where all the sum indices b, b', b'' run over the preimage $f_i^{-1}(\{a\})$, whose (random) cardinality is denoted by $K_i(a)$. Since $J_{i,i'}(b)$ are independent for distinct b, we obtain the recursion

$$\begin{aligned}
\mathrm{E}\left[J_{i,L}J_{i,i+\ell}^2\right] &= \mathrm{E}\left[K_i\right]\mathrm{E}\left[J_{i+1,L}J_{i+1,i+\ell}^2\right] + \mathrm{E}\left[K_i(K_i-1)\right]\mathrm{E}\left[J_{i+1,L}\right]\mathrm{E}\left[J_{i+1,i+\ell}^2\right] \\
&+ 2\,\mathrm{E}\left[K_i(K_i-1)\right]\mathrm{E}\left[J_{i+1,L}J_{i+1,i+\ell}\right]\mathrm{E}\left[J_{i+1,i+\ell}\right] \\
&+ \mathrm{E}\left[K_i(K_i-1)(K_i-2)\right]\mathrm{E}\left[J_{i+1,L}\right]\mathrm{E}\left[J_{i+1,i+\ell}\right]^2
\end{aligned} \tag{3.48}$$

Given the distribution of $\boldsymbol{f}_i^L$, the variable K_i is Poisson distributed with unit mean, hence we have

$$\mathrm{E}\left[K_i\right] = \mathrm{E}\left[K_i(K_i-1)\right] = \mathrm{E}\left[K_i(K_i-1)(K_i-2)\right] = 1 \tag{3.49}$$

and from previous results

$$\mathrm{E}\left[J_{i,i'}\right] = 1 \quad , \quad \mathrm{E}\left[J_{i,i'}^2\right] = i'-i+1 \quad , \quad \mathrm{E}\left[J_{i,i'}J_{i,i''}\right] = \min\{i',i''\}-i+1 \tag{3.50}$$

The recursion above becomes therefore

$$\begin{aligned}
\mathrm{E}\left[J_{i,L}J_{i,i+\ell}^2\right] &= \mathrm{E}\left[J_{i+1,L}J_{i+1,i+\ell}^2\right] + \mathrm{E}\left[J_{i+1,L}\right]\mathrm{E}\left[J_{i+1,i+\ell}^2\right] \\
&+ 2\,\mathrm{E}\left[J_{i+1,L}J_{i+1,i+\ell}\right]\mathrm{E}\left[J_{i+1,i+\ell}\right] + \mathrm{E}\left[J_{i+1,L}\right]\mathrm{E}\left[J_{i+1,i+\ell}\right]^2
\end{aligned} \tag{3.51}$$

$$= \mathrm{E}\left[J_{i+1,L}J_{i+1,i+\ell}^2\right] + 3\ell + 1 \tag{3.52}$$

which, together with the starting point

$$J_{i+\ell,i+\ell}(a) = 1 \quad , \quad \forall a \tag{3.53}$$

yields the result

$$\mathrm{E}\left[J_{i,L}J_{i,i+\ell}^2\right] = \frac{3}{2}\ell^2 + \frac{5}{2}\ell + 1 \tag{3.54}$$

We then obtain the average pairwise joint success probability as

$$\mathrm{P}\left[S_j(i,\ell) \cap S_{j'}(i,\ell)\right] = \frac{1}{N^3}\sum_{a\in\mathcal{N}} \mathrm{E}\left[J_{i,L}(a)J_{i,i+\ell}^2(a)\right] = \frac{\frac{3}{2}\ell^2 + \frac{5}{2}\ell + 1}{N^2} \ . \tag{3.55}$$

The above result on the pairwise joint success probability for guesses can be used together with the single success probabilities (3.40) to derive lower and upper bounds for the success probability of the complete attack with N_A independent guesses on the same chain section, as

$$\sum_{j=1}^{N_\mathrm{A}} \mathrm{P}\left[S_j(i,\ell)\right] - \sum_{j=1}^{N_\mathrm{A}}\sum_{j'<j} \mathrm{P}\left[S_j(i,\ell) \cap S_{j'}(i,\ell)\right] \leq \mathrm{P}\left[\cup_{j=1}^{N_\mathrm{A}} S_j(i,\ell)\right] \leq \sum_{j=1}^{N_\mathrm{A}} \mathrm{P}\left[S_j(i,\ell)\right] \tag{3.56}$$

$$\frac{N_\mathrm{A}(1+\ell)}{N} - \frac{N_\mathrm{A}(N_\mathrm{A}-1)\left(3\ell^2+5\ell+2\right)}{4N^2} \leq \qquad P_\mathrm{s} \qquad \leq \frac{N_\mathrm{A}(1+\ell)}{N} \tag{3.57}$$

Combining (3.57) with the constraint on computational power, $N_A = R_\mathrm{h}T/\ell$ and $\ell = R_k T_\mathrm{A}$, with $R_k = 1/T_k$ being the release rate for the keys, we can rewrite the above expression as

$$\frac{R_\mathrm{h}T}{N}\left(1 - \frac{1}{R_k T_\mathrm{A}}\right) - \left(\frac{R_\mathrm{h}T}{N}\right)^2 \left(1 + \frac{R_k T_\mathrm{A}}{R_\mathrm{h}T}\right)\left(\frac{3}{4} + \frac{5}{4R_k T_\mathrm{A}} + \frac{1}{2R_k^2 T_\mathrm{A}^2}\right) \leq P_\mathrm{s} \leq \frac{R_\mathrm{h}T}{N}\left(1 + \frac{1}{R_k T_\mathrm{A}}\right) \tag{3.58}$$

In the following range of values, which we consider plausible for the TESLA protocol in the GNSS OS NMA scenario:

- key length $n \geq 80$ bit;
- key release rate R_k between 0.1 and 10 key/s;
- chain duration between a few days and a few months;

and for the attack strategy:

- target attack interval T_A between a few minutes and a few days
- attacker's hashing rate R_h between 10^9 and 10^{15} hash/s
- attack computation time T between a few minutes and a few months (upper bounded by the chain duration)

the inequalities $1 \ll R_\mathrm{h}T \ll N$ and $1 \ll R_k T_\mathrm{A} \ll N$ hold, so that a much simpler expression and significantly close approximation to the attack success probability can be obtained by neglecting higher order terms

$$P_\mathrm{s} \approx \left(1 + \frac{1}{\ell}\right)\frac{R_\mathrm{h}T}{N} \approx \frac{R_\mathrm{h}T}{N} \tag{3.59}$$

showing that the success probability of the attack is bounded away from zero as $\ell \to \infty$. On the contrary, for an ideal key chain generation algorithm, where each f_i is one-to-one (i.e., a permutation

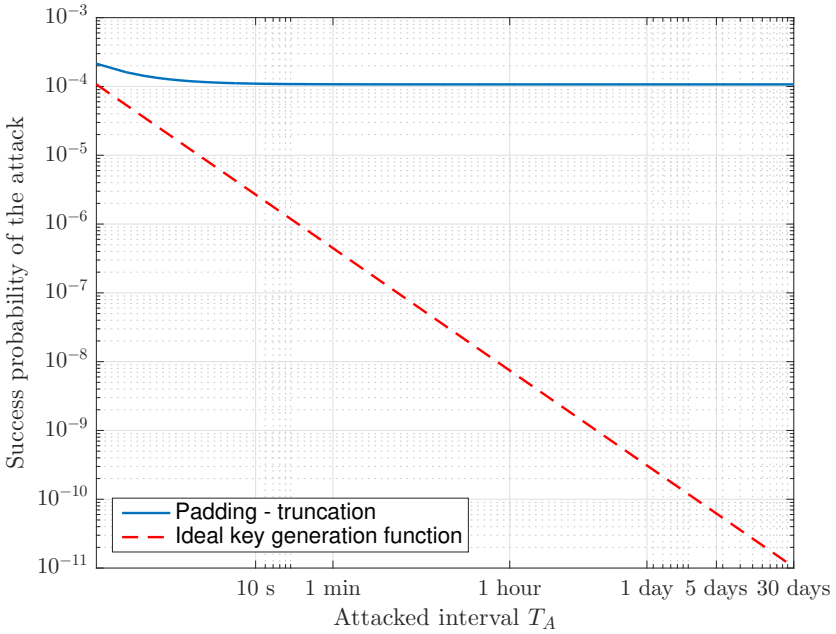


Figure 3.9: Success probability for an attack against a 80-bit key chain, with hashing rate $R_{\mathrm{h}} = 5 \cdot 10^{13}$ hash/s, and attack duration $T = 30$ days.

of $\mathcal{N}$), the only possible success for the attacker is to correctly guess the exact key used by the system, $\hat{k}_{i+\ell}^{j} = k_{i+\ell}$, which leads to a success probability

$$\mathrm{P}\left[S_j(i,\ell)\right] = \frac{1}{N} \quad , \quad P_{\mathrm{s}} = \frac{1}{\ell} \cdot \frac{R_{\mathrm{h}} T}{N} \tag{3.60}$$

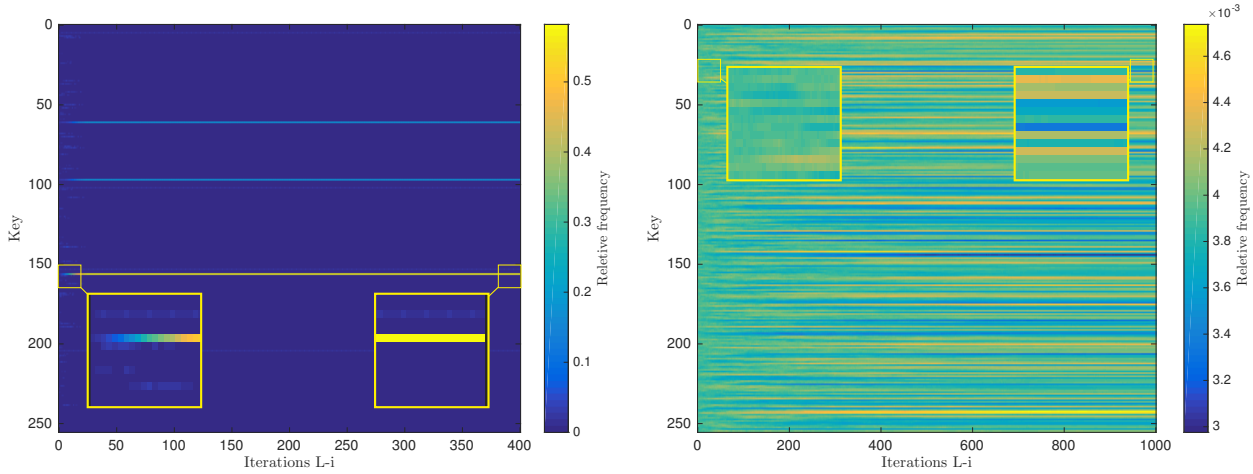
vanishing as $\ell \to \infty$. A comparison between (3.59) and (3.60) clearly shows that the use of long chains has a severe impact on the security of the scheme. Note that this would not be the case if the one-way function f was collision-free, i.e., would be modeled as a random permutation. However, this model is not realistic for most hash functions including secure one such as SHA.

An important outcome of this work is that, from a sufficiently high value of ℓ, attacking the system for a longer time does not require more computational power to obtain the same success probability, due to the non ideal key generation function. Therefore, with the same effort required to attack few iterations, it is possible to compromise the entire key chain.

Fig. 3.9 reports the success probability computed using (3.59) for system parameters that match the proposal in [50] (key length $n = 80$ bit, key release rate $1/T_k = 40$ keys every 10 s) for an attack computation time $T = 30$ days and a computational power $R_h = 5 \cdot 10^{13}$ hash/s. It is noteworthy that such a computational power can be obtained for less than US$10,000 using Bitcoin mining hardware (that it based on SHA-256) [69].

3.3.3.4 Frequency analysis

Our second analysis of the key generation function focuses on the empirical probability distribution of the output values of $\boldsymbol{f}_i^L$. Using the implementation discussed above, the number of occurrences of each output were collected at every iteration. Note that the 8-bit case is illustrated in the figures below in order to provide clarity for the reader; however, the same analysis was performed with similar results for truncation length of up to 16 bits. In the ideal case, it is expected that, at every iteration, all the values in the output space are equally probable. In Fig. 3.10a the time-invariant padding version is



(a) (b)

Figure 3.10: Frequency analysis of the key generated by a (a) time invariant and (b) time varying algorithm.

presented. It is clear that after a few iterations, the results are no longer equally distributed, having few states that are very likely and other that will not be reached. In Fig. 3.10b the analysis was repeated with time-variant padding that change at every iteration. Also in this case the results are not ideal, but the fact at every iteration the hashing function can be approximated by an independent random permutation increase the uncertainty on the output space because of the time dependent one-way function and makes difficult to predict the reachable set $\mathcal{N}_i$. The Figure represents the key chain of 1000 keys obtained starting from all the 2^8 8-bit possible starting values and for each of the 2^{16} 16-bit counter values. It is evident that, even in this case, some lines appears, showing that some output values are more probable than the others.

Fig. 3.11 shows that, as the number of performed iterations grows, the probability density function (pdf) of the output of the one-way function is less uniform. From comparison between Fig. 3.10a and Fig. 3.10b clearly stands out the effect of the time-variant algorithm on the security level of the key chain.

On the other hand, the idea outlined in [49, 50] about using 40 keys per time with the purpose of increasing the security of each satellite, is instead harming the security of the key chain. Indeed, after having iterated f for a certain number of times (which depends on the key length n) the outputs are not even equally distributed (Fig. 3.10), and consequently the uncertainty about the output value decreases considerably (Fig. 3.6). In the previous paragraph the success probability of a brute force attack without no prior information on the output space $\mathcal{N}_i$ after the i-th iteration was derived, but an attacker may be able to exploit this non uniformity, if a method to predict the pdf of $\mathcal{N}_i$ will be found, mounting a much more effective attack. This investigation is left to future work.

3.3.3.5 Design recommendations

The probabilistic model presented provides the mean collision probability and an upper bound on the expected entropy of keys generated using a key chain with padding and truncation at each iteration. This model can be used by designers of NMA schemes as a tool to optimize parameters of a one-way key chain generation algorithm for security.

The following recommendations are made for NMA schemes based on the use of one-way key chains with padding and truncation.

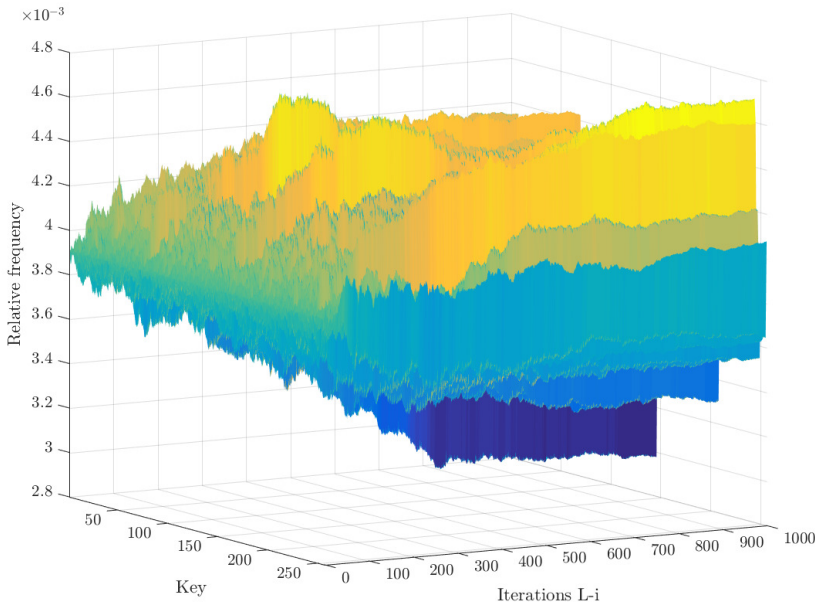


Figure 3.11: pdf of the output of the time-variant key generation algorithm for 8-bit keys with 16-bit counter.

- *Key size:* the key size should take into consideration the loss of entropy caused by the truncation. The model proposed can be used to find an adequate key size. Moreover, longer keys will be less impacted. Another aspect that should be investigated is the use of cryptographic primitives (e.g., hash function) carefully designed for smaller block size.
- *Time-variant padding:* the use of time-variant padding (e.g., GST) is strongly recommended as the addition of a time varying component will change the output set and its distribution at every iteration. This significantly limits the attacker's ability to determine the probability distribution of $\mathcal{N}_{L-i}$ and therefore provides protection against pre-computation attacks.
- *Use short chains:* the more iterations performed, the greater the loss of entropy and the higher can be the success probability of a brute force attack.
- *Minimize the frequency of key disclosure:* the beginning of the key chain (last value released) is impacted by the largest reduction of entropy. This is evident in Fig. 3.6, which illustrates that the most significant reduction occurs within 5 hours (i.e., starting from 80 bits, 15 bits of entropy are lost after approximately 16,400 iterations). It should be noted that the 10 seconds time-slot is constrained by the maximum verification delay, and ultimately by the target TTA. However, releasing a single key instead of 40 keys every slot would result in this reduction occurring in 8 days instead of 5 hours. Observe that the chain is used in reverse order of generation, and keys with much lower entropy than the ideal case are used for the majority of the chain duration. Therefore, only the last few keys to be disclosed will approach the full entropy for the key size.
- *Vary the padding at every iteration:* the use of the same padding for more than one iteration may not be ideal and could potentially lead to additional vulnerability.
- *Randomize the parameters used:* avoid the use of predictable parameters in the key chain generation algorithm. Varying parameters such as the start epoch and duration of key chains, can provide additional protection against pre-computation attacks.

- *Disclose parameters for key chain generation as late as possible:* delaying the disclosure of key chain generation parameters can provide additional protection against pre-computation attacks.

A side effect of a successful attack as described in Section 3.3.3.3 is that, even if the attack itself lasts for a limited time T_A, the victim receiver will lose the capability to authenticate the legitimate signal for the remaining duration of the key chain, since the receiver will verify received keys against a forged key. In order to avoid this side effect and recover after the attack, it is necessary to design a recovery procedure for the receiver. However, this is outside of the scope of the present work and it is left for future investigation.

3.3.4 TESLA performance analysis

In this section an analytic performance evaluation of TESLA-based NMA proposals is presented. Evaluating the actual performance achieved by an NMA scheme is a challenging task. In fact, this evaluation is influenced both by environment and receiver dependent variables. The environment where the performance are evaluated, e.g., urban or suburban, it is dependent on the particular realization of the surrounding (buildings, trees, lamppost) but also by the constellation seen in the moment of the evaluation (the GPS constellation has a periodicity of 12 hours, while the Galileo of 10 days). Hence, in order to achieve a statistical validity of the analysis this should cover a time period long enough in order to accommodate all the possible constellation combinations, and repeated in a sufficient number of environment.

Moreover, the receiver processing logic can make a big difference in the performance achieved. Indeed, it is possible to exploit NMA in different ways:

- if the receiver uses NMA only for authenticating the navigation message it can authenticate the navigation message one time for IOD, then it can stop decoding the navigation message for a certain period. This duration shall be defined based on the accepted performance degradation due to the use of deprecated navigation messages. In this case the only protection is at the data layer, so the receiver can use in the PVT computation every signal for which the authenticated ephemeris and clock correction are known.

- if the receiver uses NMA to perform some signal layer techniques to achieve signal anti-replay the receiver shall continuously decode the navigation message and shall compute the PVT using only the authenticated signals. In this case the receiver shall use only the signals in an *authenticated state* at the time of the PVT computation. The receiver logic shall define when the signal is considered in the authenticated state. For instance, due to the difficult of mounting a fully synchronized attack without cycle slip, a receiver could retain in the authenticated state a previously authenticated signal until a loss of lock or cycle slip is detected even if a single authentication verification fail, in order to increase the continuity, and to put the signal in the non authenticated state only after a series of authentication fail. Clearly this allow the attacker to influence the PVT computation up to a certain extent, that shall be adequate to the user needs. While, for safety critical application the receiver may wants to compute the PVT only with signals for which the last authentication verification was successful.

For these reasons a simplified performance assessment was performed. This analysis is not intended for provide an absolute measure of the performance of a certain NMA proposal, instead to provide a framework useful for performing a fair comparison of the techniques or to assess how the system parameters affects the performance.

Authentication Error Rate The Bit Error Rate (BER) for BPSK signals such as C/A and P(Y) can be written as [11]:

$$P_b = \frac{1}{2} erfc\left(\sqrt{\frac{C}{N_0 R_b}}\right) \tag{3.61}$$

where R_b is the bit rate, and

$$erfc(x) = \frac{2}{\sqrt{\pi}} \int_{t=x}^{\infty} e^{-t^2} dt \tag{3.62}$$

As Galileo uses a binary modulation, with FEC mechanism based on convolutional codes, assuming soft decoding, in the Additive White Gaussian Noise (AWGN) channel, and presuming perfect PLL tracking, the BER can be tightly upper-bounded by [11]:

$$P_b \leq \frac{1}{2}\left(36D^{10} + 211D^{12} + 1404D^{14} + 11633D^{16}\right) \tag{3.63}$$

$$D = e^{-\frac{1}{2R_b}\frac{C}{N_0}} \tag{3.64}$$

where R_b is the bit rate. Let us define L_d as the number of navigation data bits that the receiver needs to correctly decode in order to perform an authentication verification. Assuming that the authentication protects the Clock and Ephemeris Data (CED) of Galileo Integrity Navigation Message (I/NAV) message, $L_d = 436$ bits for each scheme, and L_m, L_k are respectively the length of the MAC and of the key for the TESLA scheme. We can write the Authentication Error Rate (AER) as:

$$\text{AER} = \begin{cases} 1 - (1 - P_b)^{L_d + L_m + L_k} & \text{with fresh data} \\ 1 - (1 - P_b)^{L_m + L_k} & \text{with data reuse} \end{cases} \tag{3.65}$$

where in the first case it is assumed that the receiver try to decode the navigation message at every authentication verification; while in the second case, after having authenticated the navigation message ones, the receiver stops decoding it, but just try to re-authenticate it with the new MAC. The schemes which perform better are those which reduce the decoding error sending a lower amount of bit both for authentication and information data, or reusing the latter.

If multiple SVs are considered, in the scheme presented in [51, 50] is proposed to use a different key all taken from the same key chain, and to release 40 key every authentication round. This allows to verify a received MAC using any of the key located before along the chain and sent from another satellite in view. In this case the performance will also depend on the number of SVs in view from the receiver, N_{SV}, the geometry of the constellation and the specific environment. For simplicity, we assume that all the SVs have a stable C/N_0. A further issue is the key release scheduler adopted in [51]. The necessary condition to have authentication is receiving at least a correct key among the previous ones (plus obviously the correct MAC). However, due to the lack of a scheduler definition, the keys assigning order within the satellite constellation is unspecified. Then, we assume that the keys order has a uniformly distributed random order among the SVs in view, averaged on the number of in view satellites (i.e., the factor $1/N_{SV}$). Thus making the assumption of reusing data, the AER can be written as:

$$\text{AER} = 1 - (1 - P_b)^{L_m} \left[\frac{1}{N_{SV}} \left(1 - \sum_{j=1}^{N_{SV}} \prod_{i=1}^{j} (1 - P_{b_i})^{L_k}\right)\right] \tag{3.66}$$

Authentication Rate Let us define x as a random variable that count the number of failures before the first authentication success. For schemes that require checking only the authentication bits, after the TTFAF, x is a geometric random variable with independent trials, each with success probability 1-AER, using $L_d = 0$. Then, the authentication rate is

$$\mathrm{AR} = \frac{1}{\mathbb{E}[x] \cdot \mathrm{TBA}} \tag{3.67}$$

where Time Between Authentications (TBA) is the time between two possibilities of authentication. If the scheme requires to decode not only the authentication data, but also the navigation data, then x has to be properly rewritten. Let us start from the single authentication trial case, then we have the probability of correctly decoding:
the navigation data

$$P_D = (1 - P_b)^{L_d} \tag{3.68}$$

and the authentication data

$$P_A = (1 - P_b)^{L_m + L_k} \tag{3.69}$$

Now, considering the option of sending 3 $MACs$ per subframe and assuming that all the $MACs$ are relative to the same data, the probability of being able to perform at least one out of the 3 authentication chances within a subframe period is:

$$P_F = P_D \left[1 - (1 - P_A)^3\right] \tag{3.70}$$

Let us define the probability $P_{B_{nk}}$ that all the authentication checks performed from $t = 0$ up to the k-th check in the n-th subframe fails. Combining the probabilities defined above, we have:

$$P_{B_{nk}} = \left[\prod_{i=1}^{n}(1 - P_F)\right]\left[1 - P_D\left(1 - \prod_{j=1}^{k}(1 - P_A)\right)\right] \qquad k = \{1, 2, 3\} \quad n = 0, 1, \ldots \tag{3.71}$$

Finally, the Cumulative Distribution Function (CDF) of the random variable x is

$$CDF(x) = 1 - P_{B_{nk}} \tag{3.72}$$

and the corresponding AR is computed by substituting the relative $E[x]$ in (3.67).

Time To First Authenticated Fix In order to analyze the TTFAF, a Markov model can be used to describe the decoding procedure: when the receiver is turned on, or when the navigation message is updated, the receiver starts decoding the message and the model keeps track of the probability of correctly decoding every single word at every instant. We will refer to the time elapsed with t. Now, defining L_{w_j} as the number of useful bits in the navigation word j, and N_w the number of authenticated words in the navigation message, it is possible to write the probability of have correctly decoded the single word W_j from a SV at time t as

$$P_{W_j}(t) = 1 - (1 - (1 - P_b)^{L_{w_j}})^{\lfloor t/T_s \rfloor} \tag{3.73}$$

where T_s is the subframe period. While, the probability of being able to authenticate a single SV_i at time t is

$$P_{\mathrm{Auth}_i}(t) = (1 - \mathrm{AER}) \prod_{j=1}^{N_w} P_{W_j}(t) \tag{3.74}$$

Let $N_{\min} = 4$, then the probability of being able to perform an authenticated fix at time t is

$$P_{\text{AuthFix}}(t) = 1 - \sum_{i=0}^{3} \binom{N_{SV}}{i} (1 - P_{\text{Auth}}(t))^{N_{SV}-i} P_{\text{Auth}}(t)^i \tag{3.75}$$

and we can finally write the CDF of the TTFAF as

$$P[\text{TTFAF} < t] = 1 - \prod_{j=0}^{t} (1 - P_{\text{AuthFix}}(j)) \tag{3.76}$$

3.4 Other security aspects of NMA

Up to now the security of the NMA schemes was evaluated taking into account only the cryptographic strength of the primitives used by the proposed protocol. In this Section a much broad view will be adopted.

Beside the cryptographic algorithm analysis, a composable security analysis is missing. In TESLA three cryptographic primitives are involved, and it was shown that one of the component, the key generation function, is not ideal. The impact of this non-ideality on the security of the digital signature and of the output of the HMAC has not been evaluated, yet. Moreover, it is missing also an end-to-end security evaluation that takes into account also the receiver processing and the hardware requirements, e.g., the oscillator stability required to ensure the time synchronization.

An important note is on the way NMA is transmitted. GNSS signals makes use of FEC coding to reduce the BER. Various types of channel coding are used by the different signal component: e.g., Galileo E1 OS makes use of convolutional code, while GPS L1C uses Low-Density Parity-Check (LDPC), while GPS C/A do not use any channel coding. SBAS L1 uses the same convolutional code of Galileo OS with different symbol rate and message length.

Hence, a certain amount of redundancy is added to the authentication message, that for its nature is random and unpredictable, reducing the entropy of the resulting message. This redundancy can be exploited by the attacker to guess part of the authentication message even prior to its actual transmission. In [74, 75] it is shown that with this attack, referred as Forward Estimation Attack (FEA), it is possible to generate a signal that will be correctly decoded by the victim receiver with up to 116 ms of advance for Galileo OS and up to 100 ms for GPS L1C, inducing ranging errors on the order of thousands of kilometers. This attack is harmless if the NMA scheme is not used to protect the ranging, indeed it does not allow to modify the navigation message, but may impair the security of the ranging protection. The detection strategies based on the symbol unpredictability (e.g., SCER, 4.4) with the FEA attack will achieve lower performance in terms of detection capability. If the authentication symbols are not directly used for spoofing detection, but rather it is used the information carried by them (e.g., delayed disclosure of secret spreading code used), the FEA does not affect the detection capability, as long as the disclosure delay is bigger than the maximum estimation interval.

These works show that, in order to maximize the security of the scheme, the authentication message should be broadcast without FEC and the receiver should use hard-decoding. On the other hand, this will clearly reduce the performance in terms usability, increasing the AER.

Moreover, while for pseudo-packetized message format such as the GPS Civilian NAVigation (CNAV) and the SBAS L1 message in which the order of the message is in general non predictable, in the case of Galileo OS the transmission time of the authentication message is predetermined. Even this information can be exploited by an attacker. [76] shows that it is possible to build a smart jammer that, knowing the coding scheme used and the exact position of the authentication bits in the message, minimize the amount of energy that must be transmitted in order to disrupt the correct reception of

the authentication data. It is shown that this kind of smart jammer is not detected by traditional metrics such as the C/N_0 or AGC based, due to the short pulse duration and the effect of the channel coding and interleaving. In this case, the attack is a DoS, where the goal is to prevent the victim receiver from performing authentication, minimizing the probability of being detected.

The authentication provided by NMA can enable new services, such as pay per drive. In this context, it is likely that the attacker will be the user itself trying to pay less for the road tolling or to the insurance company. In this case the smart jammer can probably be the user that will render useless the authentication scheme enforced by the authority or the service provider, without being detected. To prevent this kind of attacks, smart detection strategies and signal layer mechanism should be employed.

The case of the self spoofer is the most challenging one, because in this case the attacker can easily have access to the receiver and the antenna. In order to prevent tampering of the receiver, a series of countermeasure shall be put in place [31]. Any piece of the receiver shall be authenticated to prevent that the attacker can alter the PVT reported: from the antenna, in order to prevent that the attacker plug a signal generator; to the clock, to prevent trivial alteration of the oscillator frequency (e.g., due to the temperature); and digitally signing each intermediate step in the software to avoid hacking.

Instead of building tamper resistant device, in [77] a way to fingerprint the receiver is discussed without the use of cryptography. The technique proposed uses the characterization of the local oscillator to detect the tampering or the replacement of the receiver. Heuristic detection scheme such this, have the advantage of being simple to implement, but the actual performance against sophisticated attack shall be evaluated.

From these analyses it appears clearly that the NMA alone is not able to protect the ranging level and should be coupled with a signal layer technique. Conversely, ranging attacks such as meaconing, do not affect message integrity, therefore the NMA verification can be successful even in presence of attacks. Thus, it is not optimal to design a data authentication mechanism in order to provide signal authentication, sacrificing the security of the mechanism itself. For this reason in the next Section a novel NMA scheme designed with the main goal of providing data authentication will be presented.

3.5 SigAm: a digital signature amortization scheme for the GNSS navigation message

In this Section a novel NMA scheme, named SigAm, will be presented that improves on both security and bandwidth efficiency with respect to currently proposed NMA schemes and does not require time synchronization.

As highlighted in the previous Section, chaining is a clever solution for the GNSS context, since it mitigates the demand for communication and therefore, it is worth exploiting. Beside TESLA, one way chains are exploited in a different way with a technique called digital signature amortization. This technique exploits the one-way function not to generate a key chain, but to build a signature chain, that provides the authentication of the message itself.

3.5.1 Classical digital signature amortization

The concept of digital signature amortization relies on extending the length of the message authenticated by a digital signature, while allowing intermediate verification. The construction of the signature chain can be summarized as:

The traditional version of digital signature amortization [78] is not adequate to the use in GNSS, mainly for the lack of loss tolerance. Indeed, the loss of some packets in the signature chain may lead to the inability to perform the authentication. A variant of this scheme, called EMSS [62], introduces loss tolerance at the price of multiple digests being transmitted along the same message, and is therefore

1: Initialize the authentication tag H_M, where M is the length of the chain
2: Initialize $m = M - 1$
3: **while** $m \geq 1$ **do**
4: Concatenate the message D_m with the authentication tag computed at the previous iteration H_{m+1}
5: Compute the authentication tag H_m
6: $m = m - 1$
7: **end while**
8: Sign the digest H_1 with the sender private key k_{priv}
9: Compute the signature of H_1 to obtain the signature of the whole chain

Algorithm 1: Digital signature amortization algorithm.

itself not particularly suited to GNSS because of bandwidth limitations. In the following we introduce several modifications to the basic scheme to suit the NMA problem.

3.5.2 Binding each chain to a single IOD

A defining characteristic of the GNSS navigation message is that it does not change frequently, e.g., the ephemeris and clock correction data do not change for an Issue Of Data (IOD), that can last for several hours. During this time, it is not necessary for GNSS receivers, once the data are correctly decoded, to demodulate the navigation message again. The first proposed adaptation exploits this characteristic. Binding the signature chain duration to the IOD duration, allows to demodulate the data only once and achieve loss tolerance. An important property introduced by this modification is that messages in each IOD are authenticated by a different digital signature. This allows to perform data authentication relying on the security of standard, well known, cryptographic primitives that are considered secure by the cryptography community. This is an advantage with respect to TESLA, where in order to save bandwidth, the message is authenticated by a very short MAC, e.g. 10-30 bits.

3.5.3 Improving data anti-replay capability

The proposed scheme also offers a data anti-replay capability through the use of the one-way chain. In fact, an attacker may replay expired authentic navigation message to degrade the accuracy of the computed PVT in what is commonly referred to as data replay attack. The proposed concept has the advantage that, after the first authentication, the receiver could check if the previously authenticated navigation message is still valid, with the reception of a single authentication tag that is shorter than a digital signature or the key/MAC pair for TESLA.

An additional protection can be achieved by introducing a time parameter in the signature computation. Ideally, this parameter should change at every signature repetition, but in this case the receiver is not able to accumulate the signature chunks from different repetitions over multiple sub-frame. A trade-off shall be found, and a possible solution can be to update the signature a few times within the IOD validity, e.g., every 15-30 minutes.

Finally, note that although SigAm itself is not designed to provide signal anti-replay capability, it can be coupled with a signal layer technique, in which the SigAm tags be used as an authenticated source of entropy, such as SCER detection [30] or Spreading Code Encryption (SCE) [79] key generation.

3.5.4 Guaranteeing authentication continuity across IODs

When the navigation message changes, the receiver continues to compute the PVT using the old authenticated ephemeris until it is able to verify the authenticity of the new one. In order to allow a seamless handover between two IOD, thus increasing the continuity of the authentication, we introduce the concept of overlapping authentication chains, as illustrated in Fig. 3.12.

When the IOD changes, the system can continue to broadcast the authentication tags taken from the old signature chain, together with the new navigation message and the corresponding digital signature. After a certain time, it can start broadcasting authentication tags taken from the new chain. If the transmission of navigation data, signature and authentication tags starts synchronously, after the chain transition the receiver will not be able to perform any verification, but it can still navigate using the old authenticated navigation data. For the purpose of data authentication this is fine. However, if SigAm is intended to be coupled with a signal layer technique for achieving signal anti-replay capability, then this mechanism loses its authenticated source of entropy.

On the contrary, if the transmission of authentication tags taken from the new signature chain is delayed with respect to the navigation message change, the signal level authentication technique can continuously verify the authentication tags. From the data layer perspective, the receiver is able to use the data-anti-replay capability, verifying the freshness of the past navigation data, while the authentication of the new one is postponed.

In setting the overlapping time, a trade-off has to be sought. The later the first authentication tag of the new chain is transmitted, the more probable the receiver is to perform a seamless handover increasing the signal layer authentication continuity for receivers in challenging environments. However, the delay will force all receivers, even those operating in good channel conditions, to extend the time required to apply the new corrections. On the other side, the shorter the overlapping, the shorter the Time To First Authentication Fix (TTFAF) will be as well. Indeed, if a receiver is turned on after the IOD change, it will not be able to verify the navigation data until the system starts broadcasting authentication tags taken from the new chain.

It is worth observing that shortening the overlapping time below the best case Time To First Fix (TTFF) is useless, because the receiver will not be able to authenticate the data anyhow. The amount of overlapping is a system parameter that can be configured finding a good trade-off between the performance experienced by users in good conditions, those in challenging environment, and the user requirements.

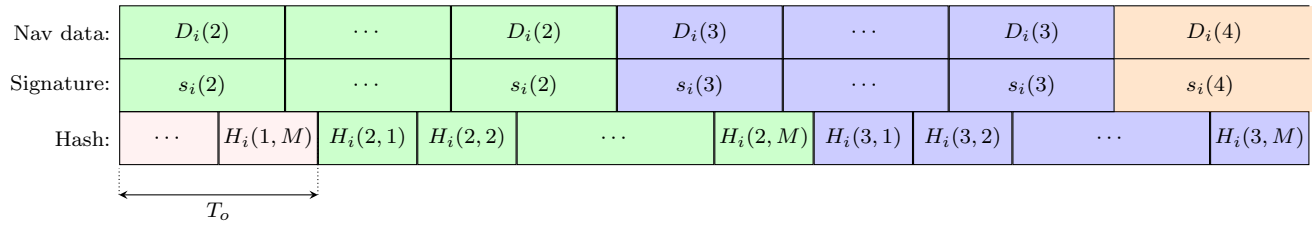


Figure 3.12: Example of SigAm frame format for simultaneous broadcast of navigation data, signature and hash digests.

3.5.5 Algorithm formulation

The proposed authentication scheme can be formalized by dividing it into four operations: computation of the signature chain, transmission, reception and verification. Let us define the one-way function h as:

$$h(D, H) \triangleq \text{trunc}(\text{HASH}(D \parallel H \parallel t), \ell_H) \tag{3.77}$$

where HASH is a cryptographically secure hash function, e.g., SHA-224 or SHA-256, and t is the system time, e.g., the Z-count or the GST, that is used as a counter to prevent pre-computation attacks, and trunc denotes the truncation of x to ℓ_H bits. Note that the above definition of $h(D, H)$ is not the only possible solution and can be adapted using other formulations such as keyed hash functions (e.g., HMAC [62]). Denoting the part of the navigation message that changes infrequently, e.g., ephemeris and clock correction data, as $D_i(n)$ where i is the index of the SV_i broadcasting the message and n is the IOD, we can write the four operations as:

1. Signature:

$$H_i(n, M) : \text{random and uniform} \tag{3.78}$$
$$H_i(n, m) = h(D_i(n), H_i(n, m+1)) \ , \ \ 1 \leq m < M \tag{3.79}$$
$$s_i(n) = \mathbb{S}(k_{priv_i}, [D_i(n) \parallel H_i(n, 1) \parallel t]) \tag{3.80}$$

2. Transmission:

$$\mathbf{H}_i(n) = [H_i(n, 1), H_i(n, 2), \ldots, H_i(n, M)] \tag{3.81}$$
$$x_i(n) = [D_i(n), s_i(n), \mathbf{H}_i(n)] \tag{3.82}$$

3. Reception:

$$\hat{\mathbf{H}}_i(n) = \left[\hat{H}_i(n, 1), \hat{H}_i(n, 2), \ldots, \hat{H}_i(n, M)\right] \tag{3.83}$$
$$\hat{x}_i(n) = \left[\hat{D}_i(n), \hat{s}_i(n), \hat{\mathbf{H}}_i(n)\right] \tag{3.84}$$

   where the $\hat{x}(\cdot)$ notation accounts for possible forging attacks, illegitimate modifications or channel induced errors.

4. Verification: check if

$$u = \mathbb{V}\left(K_{publ_i}, \left[\hat{D}_i(n) \parallel \hat{H}_i(n, 1) \parallel t\right], \hat{s}_i(n)\right) \tag{3.85}$$
$$\hat{H}_i(n, m-1) = h(\hat{D}_i(n), \hat{H}_i(n, m)) \ , \ \ 2 \leq m \leq M \tag{3.86}$$

   Accept the signature if $u = \textit{true}$ otherwise reject. Accept $\hat{H}_i(n, m)$ only if applying the one-way function $\hat{H}_i(n, m-1)$ is obtained.

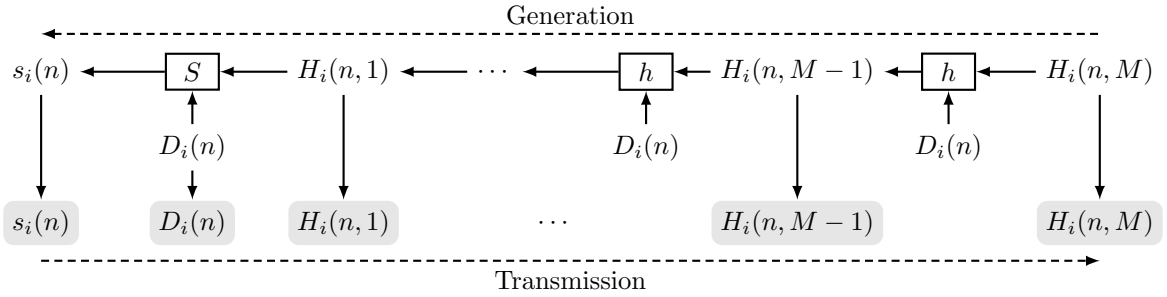


Figure 3.13: NMA based on signature amortization.

3.5.6 Attestation to third party

Observe that, being entirely based on asymmetric cryptography, SigAm can enable attestation to third party, for instance service providers. This is the security service that allows a receiver to prove the authenticity of a message to a third entity, and it can only be provided by asymmetric cryptography. Indeed, in symmetric cryptography, the secret key is shared by both transmitter and receiver, thus both can sign a message. Instead, in the asymmetric paradigm, only the sender knows the private key that allows to compute the signature.

In fact, the authentication provided by NMA can enable new services, such as pay per drive. In this context, it is likely that the attacker will be the user itself trying to pay less for road tolling or insurance fees. Service provider may require the user to attest that the reported PVT was computed using the authentic navigation message broadcast by the system. For this reason, attestation shall be a feature of the NMA scheme. SigAm, just as other digital signature based NMA schemes, offers the attestation capability, while this is not possible with TESLA based NMA schemes, due to the use of symmetric cryptographic primitives to authenticate the navigation message.

3.5.7 Comparison with other schemes

With SigAm, when a receiver is switched on, in order to perform the first authentication, it needs to demodulate the navigation data plus the digital signature of the chain and at least one of the authentication tags. This represents a disadvantage, as it may increase the TTFAF, with respect to both plain digital signature, requiring the additional reception of the authentication tag; and to TESLA where due to the long key chain (i.e., the chain handover process is not frequent), usually the receiver needs just a MAC and a key in addition to the navigation message to perform the first authentication.

However, with respect to TESLA, SigAm has the advantage of immediate authentication: in TESLA after the reception of a MAC, the receiver needs to wait for the reception of the corresponding key, while in SigAm when an authentication tag is received it can be immediately verified. This is because the values from the one-way chain are not keys used to compute authentication tags, but authentication tags themselves. For this reason, there is also no need for time synchronization between the receiver and the system. Indeed, even if the clock bias of the receiver is large enough to allow the attacker to get the authentication tag before the victim receiver expects it, this could only be used to modulate a spoofed signal, potentially changing the ranging. However, as already discussed, protecting the ranging is not a target of NMA. Due to the lack of time synchronization requirement, SigAm results in a more flexible scheme with respect to TESLA, allowing more freedom in the configuration of the system parameters, without imposing delay in order to maintain the security of the scheme. Furthermore, there is no need to have *slow authentication* MACs [50] in order to support a secure initialization of receivers at start-up, that might have a significant clock uncertainty.

The removal of the time synchronization constraint, together with self-authenticating tags and standard digital signatures, results in a much simpler receiver state machine with respect to TESLA, leading to an easier receiver implementation and validation.

3.5.8 System parameters

The proposed scheme has many degrees of freedom that can be tuned by the system designers to find the optimal trade-off. Precisely, the main system parameters are:

- *Signature scheme:* the algorithm used for the computation of the digital signature. Due to the limited bandwidth available, the EC variants of digital signature schemes such as DSA and Schnorr should be preferred. These schemes, for a security level of 80 bits, require a signature length of 320 bits and a public key size of 160 bits. If it is acceptable by the use case to update

the public key through an aiding channel once every several months or years, an interesting alternative is ETA, that for the same security level yields a shorter 240-bit signature.

- *Security level of the signature:* based on the desired level of security and the selected signature scheme different signature length and key size are obtained.
- *Signature refresh time:* validity period of a digital signature. In order to protect from data-replay attacks the digital signature can be periodically updated.
- *Signature repetition rate:* the digital signature shall be repeatedly broadcast in order to allow receivers to switch on at any time and to collect it, possibly over multiple repetitions in case of decoding errors. The repetition rate will influence the minimum time required for a receiver to perform the first authenticated fix (TTFAF). Moreover, will influence the continuity of authentication in the transition between different IODs.
- *One-way function:* the function used to generate the one-way chain, e.g., a cryptographic hash function. Different functions can be used and could have different performance.
- *Authentication tag length:* in order to reduce the bandwidth requirement, it is possible to truncate the output of the one-way function in a similar way as is done with TESLA. Although this construction potentially suffers from the same entropy reduction of TESLA, see Section 3.3.3, in SigAm the effect is less pronounced due to the short chain length (one IOD) and the lower authentication tag refresh rate.
- *Update rate of the authentication tag:* the time between the broadcast of two consecutive authentication tags determines the TBA that is the minimum time required by the receiver to perform a new authentication verification. If the scheme is intended to provide only authentication of the navigation message, this update rate can be low (e.g., one new authentication tag each sub-frame). Lower update rate will increase the TTFAF. If the scheme is intended to provide a source of unpredictability to enable anti-replay capability at the signal level, it is possible to increase the update rate. An advantage with respect to TESLA is that, being more bandwidth efficient and not requiring time synchronization, it is possible to achieve a shorter TBA. The smaller the update rate, the smaller is the bandwidth requirement.
- *Overlapping time:* a longer overlap interval increases the continuity of the authentication service, but limits the TTFAF.
- *Channel coding:* it is possible to exploit channel coding techniques applied directly to the authentication data in order to reduce the BER. The use of FEC and interleaving shall be evaluated based on [74] and the security assumptions used in the design of the NMA scheme.
- *Key management scheme:* beside the NMA scheme, a key management scheme is required. The rekeying rate and key revocation procedure shall be designed in accordance with the desired security level and shall be fit within the bandwidth devoted to authentication for key management purposes.
- *Number of SVs authenticated by a signature chain:* the navigation data authenticated by the signature chain can belong to a single SV or to a group of SVs. The former choice yield a higher security due to the independence among different chains. It is also possible to cluster a group of SVs, e.g., that serves geographically the same area, and to compute a group signature chain.

In the following, three example configurations of SigAm will be presented. All these configurations are making use of the EDBS channel of Galileo E1B service, only for the sake of comparison with the

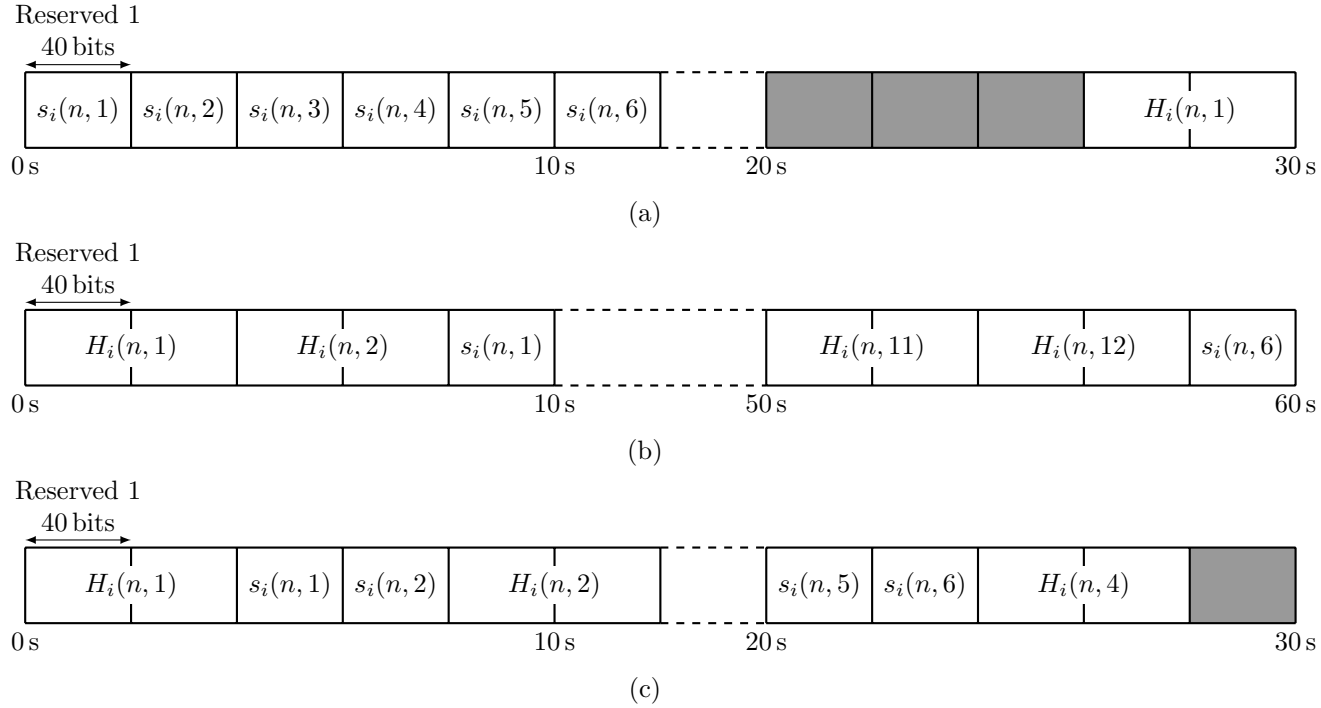


Figure 3.14: Three possible subframe allocations for SigAm in the EDBS Galileo OS channel. The 240-bit ETA signature is split into six 40-bit words $s_i(n) = [s_i(n,1), \ldots, s_i(n,6)]$, repeatedly transmitted and interleaved with 80-bit digests, which take two words each: (a) version 1, with 2.7 bit/s devoted to digest transmission and 8 bit/s for the signature; (b) version 2, with 16 bit/s for digests and 4 bit/s for the signature; (c) version 3, with 10.7 bit/s for digests and 8 bit/s for the signature.

current TESLA proposals, but can be trivially adapted to other sub-frame structures. In addition, 80-bit authentication tags are assumed. The three variants are illustrated in Fig. 3.14. The first variant (v1), Fig. 3.14a, intended for pure NMA service leads to the minimal bandwidth requirement. In this case a single authentication tag is broadcast every sub-frame together with the repetition of a full digital signature.

The second variant (v2), Fig. 3.14b, is intended to be coupled with a signal layer mechanism to enable signal anti-replay capability, and aims to minimize the TBA and maximize the symbol unpredictability. For this reason, the digital signature spans two consecutive sub-frames, and all the remaining bandwidth is devoted to the broadcast of authentication tags. Assuming the use of ETA signature, this leads to a bandwidth requirement of 4 bps for the signature broadcast. As an example, in the case of the EDBS channel, that has an equivalent bandwidth of 20 bps, this configuration allows to broadcast up to 6 authentication tags per sub-frame achieving a TBA of 5 seconds.

The third variant (v3), Fig. 3.14c, represents a trade-off between the previous two, and aims to minimize the TTFAF. The digital signature is broadcast every sub-frame, and all the remaining bandwidth is devoted to the broadcast of authentication tags. Again, by assuming ETA signatures and the EDBS channel, this configuration allows to broadcast up to 4 authentication tag per sub-frame achieving a TBA of 7.5 seconds and leaving 40 spare bits that can be used for service information such as parameters reconfiguration and key management/revocation.

3.5.9 SigAm security analysis

SigAm is designed to be a NMA scheme, thus its security shall be evaluated in term of ability to provide data authentication and integrity protection. Ranging protection is not one of the main goals

of NMA, so attacks that are trying to modify the ranging measurement are not considered. NMA can be useful in two different scenarios. In one case the user itself may attack his own receiver, wishing to report a wrong PVT solution to a third entity. As already discussed, a NMA mechanism shall allow to prove to this third party that the navigation message was originated by the system, and this can only be achieved through asymmetric schemes, with SigAm meeting this requirement. In the second scenario the attack model foresees an attacker that is trying to have the victim receiver accept a forged navigation message as authentic, or to replay expired data in order to degrade the performance of the victim receiver. In the following we discuss how SigAm is robust against this class of attack.

SigAm is based on the composition of two cryptographic primitives: digital signature and one-way function. Each navigation message is authenticated by a digital signature, so an attacker that aims at modifying the navigation message shall compute a new digital signature. If the digital signature scheme used is secure, SigAm achieves data authentication. On the other hand, the one-way chain is only used to provide the ability to re-authenticate the data after the first verification with the digital signature. The attacker that tries to perform a replay of the data must be able to find a one-way chain of authentication tags that leads to the root tag, authenticated by the digital signature.

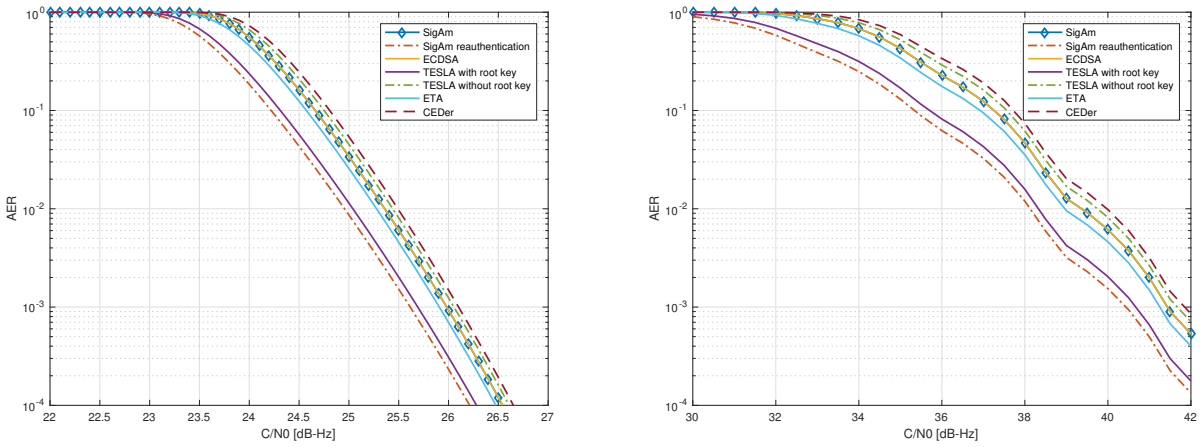
The security evaluation of one-way chains presented in Section 3.3.3 remains valid even for SigAm. The authentication tag size and its refresh rate shall be dimensioned accordingly. The low hashing rate proposed (from 2 authentication tags per minute up to 12 authentication tags per minute depending on the configuration) and the short chain length (limited to one IOD) lead to a much smaller entropy reduction with respect to the proposal in [50]. This makes it more difficult to attack the SigAm one-way chain with respect to the TESLA key chain with the currently proposed parameters (240 key per minute and chain that last for months) [50]. This justifies the use of 80-bit authentication tags. Depending on the scheme configuration and on whether it is used in combination with a signal anti-reply mechanism, the tag length can be tuned. Indeed, if SigAm is only used as an NMA scheme the 80-bit length represents a conservative choice and it may be possible to shorten it in accordance with the maximum allowed success probability of data-replay attacks.

Irrespectively of the tag length, the maximum continuous attack time is upper-bounded by the signature chain length plus a signature refresh time. The attack can be divided in two phases, based on whether it is taking place during the validity period of the navigation message or after its expiration. In the first phase, the attack is not directed against data layer but rather freely modify the ranging; and he is able to also deceive any authentication mechanism working at signal layer, that makes use of SigAm as a source of entropy. After the expiration of the IOD, the attack becomes a data replay. The maximum duration of this attack is the signature update time. Indeed, if the attacker is able to find a one-way chain that lasts for more than this time, he will not be able to compute a valid digital signature to extend the validity, and the lack of its reception will result in the dropping of the signal by the receiver.

It is worth noting that the security of the scheme is not dependent on the time synchronization, as the digital signature is completely independent of the time, while the authentication tags shall be considered unpredictable only before their transmission, but after being broadcast they cannot be used to sign arbitrary navigation messages, as happens for TESLA key chain.

Another difference with the current TESLA proposals is that in SigAm, due to its bandwidth efficiency, it is possible to use a different signature chain for each SV. This allows to introduce independence between the SVs, and an attacker that aims at modifying the navigation message of multiple SVs at the same time, is required to simultaneously compromise multiple private keys.

Finally, the security analysis of the scheme shall also take into account a key management scheme. For instance, more frequent updates of the public key used for verify the signatures, allows to shorten the keys to be used guaranteeing the same security level.



(a) (b)

Figure 3.15: AER comparison in (a) AWGN channel and (b) 2-state suburban LMS channel model.

3.5.10 SigAm performance evaluation

The robustness of the NMA scheme against channel errors, in the absence of malicious threats, is a critical point. This is usually measured in terms of AER, which is the probability of failures in authenticity verification due to channel impairments.

The scheme performance depends on the amount of data that have to be correctly received in order to verify message authenticity. Digital Signature based schemes requires the navigation message and its signature. TESLA based NMA requires the navigation message, root key and its signature, and at least one MAC and the key used by the sender to compute it. In order to increase the efficiency, usually TESLA based NMA makes use of long key chains. In this way, receivers are very likely to already hold the root key and its signature at the time of switching on. SigAm requires the navigation message, the signature and at least one authentication tag. In SigAm the signature chain lasts for one IOD, thus it is more likely that receivers do not have the current signature at start up.

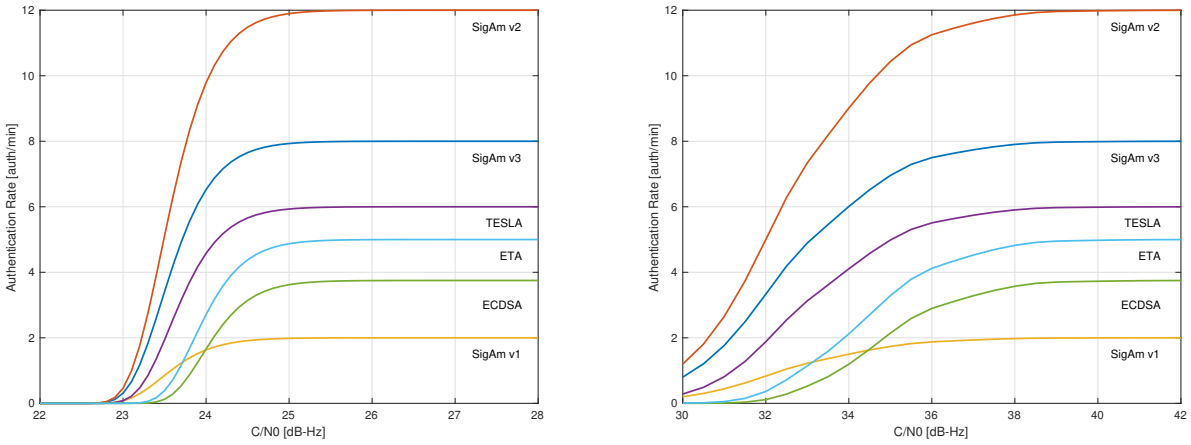
The AER can be computed as:

$$\text{AER} = 1 - (1 - P_b)^{L_d + L_a} \tag{3.87}$$

where L_d is the number of navigation bits (we have assumed four Galileo I/NAV words), L_a is the amount of bits required the specific NMA scheme, and P_b is the bit error probability. For Galileo E1B signal P_b in the AWGN channel can be computed using (3.63). In order to assess a more realistic scenario, a sub-urban 2-state LMS model with user velocity of 50 km/h and satellite elevation of 40 degrees [80] is considered. The results are shown in Fig. 3.15a and Fig. 3.15b respectively. The dashed line represents the Clock and Ephemeris Data (CED) error rate, which can be seen as a baseline in terms of Carrier to Noise (C/N0) ratio required to meet the desired BER. We use this line as a reference in the comparison with NMA schemes. It is possible to see that all the authentication schemes lie below the CED curve. Thus, this is the dominant term in the C/N0 requirement to enable authenticated navigation for autonomous users.

SigAm is assumed to use 80-bit authentication tag and 240-bit ETA signature; and it is compared with:

- TESLA using 80-bit keys and 10-bit MACs (plus 16 bit of MAC header) [50] and 320-bit ECDSA signature.
- Plain 240-bit ETA signature.



(a) (b)

Figure 3.16: Authentication Rate comparison in (a) AWGN channel and (b) 2-state suburban LMS channel model.

- Plain 320-bit ECDSA signature.

The size of the parameters above is chosen to approximately match the same 80-bit security level of the cryptographic primitives. It can be seen that the best performing scheme is TESLA when the root key is already known, while in the case that it needs to be acquired the scheme becomes the least performing. ETA signature is the second performing scheme on this performance indicator. SigAm and ECDSA perform equally for the proposed parameters. Indeed, the amount of authentication bits required by ETA plus an authentication tag equals an ECDSA signature. Nevertheless, it can be seen that for re-authentication of a navigation message, for which the digital signature is already known, SigAm outperforms also TESLA based NMA schemes (Fig. 3.15a, Fig. 3.15b).

The second performance indicator analyzed is the Authentication Rate (AR) that is the number of successful authentication that a receiver is able to perform per unit time, and is the reciprocal of the Mean Time Between Authentications (MTBA). The AR can be computed as:

$$\text{MTBA} = \frac{\text{TBA}}{1 - \text{AER}} = \frac{1}{\text{AR}} \tag{3.88}$$

Note that AR evaluates the performance after the first authenticated fix, when the receiver has already received the navigation message and the digital signature of the root authentication tag or the root key, in SigAm and TESLA respectively, so the AER in the above equation is defined accordingly. In this metric ECDSA is the less performing scheme, due to the higher bandwidth requirement. Indeed, the 320-bit signature limits the number of signatures broadcast over the EDBS channel to less than 4 per minute. ETA with its shorter signature is able to raise this rate to 5 authentications per minute. TESLA when using 80-bit keys plus 10-bit MACs can achieve 6 authentications per minute. Due to its bandwidth efficiency, SigAm is able to achieve 8 or 12 authentications per minute in the second and third proposed variants for the sub-frame configuration, respectively.

3.6 Application of joint cryptographic verification and channel decoding to GNSS

In order to limit the AER, apart from inserting redundancy or using FEC, it is also possible to use some special verification strategy. In [81, 82] it is proposed a technique that makes uses of soft output

decoding to reduce the AER.

Suppose that the transmitter sends a message $\boldsymbol{m} = (m_1, m_2, \ldots, m_n)$ and an authentication tag $\boldsymbol{a} = (a_1, a_2, \ldots, a_k)$, like a MAC. A soft output decoder produce as output the Log-Likelihood Ratios (LLRs) $\lambda(m_i)$ and $\lambda(a_i)$, defined as follows:

$$\lambda(m_i) = \log\left(\frac{P(m_i = 1)}{P(m_i = 0)}\right), \quad i = 1, 2, \ldots, n \tag{3.89}$$

The decoder is followed by a Maximum Likelihood (ML) decision performed bit by bit, assuming

$$\hat{m}_i = \begin{cases} 1 & \text{if } \lambda(m_i) \geq 0 \\ 0 & \text{if } \lambda(m_i) < 0 \end{cases} \tag{3.90}$$

and the same for the MAC part. Let we define the subset $\mathcal{S}_j$ with $j = 0, 1, 2, \ldots, i_{max}$, as the subset of the j bits whose LLRs are closer to 0. Defining a *valid pair* as a pair which satisfy $\boldsymbol{a} = \text{MAC}(\boldsymbol{m})$, the maximum likelihood of a valid pair is the one that minimize $f(\boldsymbol{m}, \boldsymbol{a}) = \sum_i |\lambda(m_i)| + \sum_j |\lambda(a_j)|$. The receiver shall set a threshold f_{th} and accept only the messages that lead the cost function $f(\boldsymbol{m}, \boldsymbol{a})$ to exceed this threshold. The algorithm is detailed in Algorithm 2.

```
1: Compute the LLRs
2: Set i = 0
3: if i > i_max then
4:     Output = FAILURE
5: end if
6: Generate m̃ from m̂ by flipping the bits in S_i
7: if ã = MAC(m̃) then
8:     Compute f̃ = f(m̃, ã)
9:     if f̃ > f_th then
10:        i = i + 1
11:        goto 4
12:    else
13:        Output = m̃ and the reliability value f̃
14:    end if
15: end if
```

Algorithm 2: Joint decoding and MAC verification.

It is worth nothing that this algorithm reduces the security level offered by the MAC for the same length [83]. This is because the algorithm will accept also message with at most i_{max} incorrect bits, if the wrong bits are those with the lower LLRs. Furthermore, if an adversary can inject noise in the transmission can lower the SNR on a set of bits of the message, those bits that the algorithm will try to flip due to the lower LLRs. Suppose that the maximum number of iteration is $i_{max} = 2^L - 1$, the attacker can degrade the SNR of L bits and eliminate 2^L possible pairs with a single incorrect guess. For this reason, if this scheme is used the MAC should be longer of L bits.

Fig. 3.17 shows a performance evaluation performed simulating a BPSK modulation over AWGN channel and comparing the AER achieved by traditional hard decoding and by joint decoding. The message length n was assumed as 500 bits and the tag length $k = 80$ bits. The maximum number of errors accepted was set to 5, thus the comparison was made with a tag length of $k' = 85$ bits when joint decoding is used. It is possible to see that joint decoding can achieve better decoding performance.

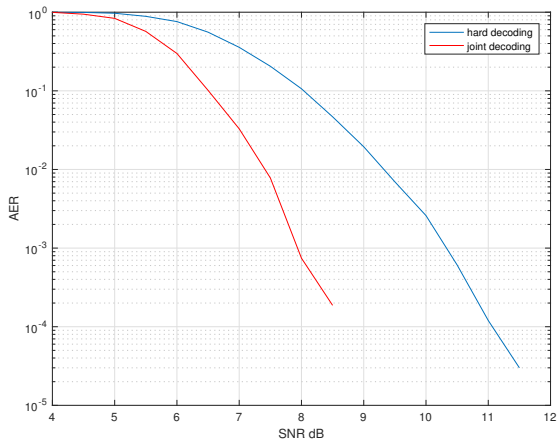


Figure 3.17: Comparison of AER of conventional and joint decoding and verification.

3.7 Data authentication for SBAS

The European Commission has launched a specific action to develop SBAS authentication solutions, the EAST (EGNOS Authentication Security Testbed) Project for the European Commission, performed by Qascom, GMV and University of Padova.

To date, the SBAS service is available as a data stream modulated with a Coarse/Acquisition Pseudo-Random Noise (PRN) at 1575.42 MHz in the L1 band. The service transmits 500 symbols per second, obtained by the convolutional encoding of 250 data bits. In the near future, a new generation of SBAS navigation payloads will be operative, and the SBAS providers are developing a new Dual Frequency Multi Constellation (DFMC) standard. In this standard, a new SBAS signal in L5 could incorporate additional data that will be available to dual-frequency users. From a signal component perspective, the current configuration and the future improvements indicate four candidates for the implementation of SBAS authentication: SBAS L1 I-channel, L1 Q-channel, L5 I-channel and L5 Q-channel [84, 85].

- SBAS L1 I-channel indicates the current L1 SBAS service. The implementation of a new service within the consolidated configuration is easily accessible to legacy uses; however, it could face backward compatibility issues.

- SBAS L1 Q-channel indicates an additional SBAS signal transmitted in quadrature with the current L1 SBAS signal (not available today). The Q component could carry the authentication service, ensuring compatibility with L1 legacy users, with no impact on the message flow. The In-Phase signal power would be reduced in case it is not possible to increase the transmitted power and the new channel have to share the power with the current signal component.

- SBAS L5 I-channel indicates the signal component that will carry the future dual frequency SBAS service. This option could be a promising way of implementing authentication since SBAS L5 standards are being developed at the moment and then there is a window of opportunity to consider the standardization of authentication in SBAS L5 I-channel at international level.

- SBAS L5 Q-channel indicates an additional SBAS signal transmitted in quadrature with the future L5 SBAS signal. Compared to the L1-Q, the limitations can be relaxed as it can rely on new devices at the space segment.

As already discussed, cryptographic authentication schemes can be divided into two main classes: symmetric or asymmetric. In symmetric schemes, the same key is shared between sender and receiver, and both of them can perform the same operation. Low computational effort and smaller key size are the main benefits of the symmetric schemes: however, the key must be kept secret and this has a significant impact on the receiver design (requires tamper resistance [31] to protect from key leakage) and on the key management.

Thus, many of the protocols taken into account in the scope of the SBAS authentication are based on asymmetric crytpoprimitives. According to the ENISA report [36], a good candidate is the Elliptic Curve (EC) variant of the Schnorr signature. Due to the high level of maturity and the widespread use of DSA, the ECDSA algorithm is considered. DSA is considered due to its short signature length, but has the drawback of a much longer public key than ECDSA, so its use is expensive in terms of key management. As contrary, even if recommended by ENISA, schemes such RSA-based signatures, have been discarded due to their long signature and public key length.

A more recent proposal, Efficient and Tiny Authentication (ETA) [47], has been also considered.

Beside digital signatures, approaches based on one-way functions that are currently being considered for GNSS NMA have been considered. In this class lay both TESLA (Section 3.2.2) and SigAm (Section 3.5), which was discarded since it requires previous knowledge of the message to compute the signature chain and this is not feasible in SBAS context.

Once the selection of the cryptographic schemes has been performed, the design process moves to identify how to broadcast the authentication data. The most intuitive solution would be to insert the authentication data in the spare bits of the current SBAS messages. However, the SBAS ICDs, both the standardized L1 and future L5, do not leave enough space available. There are then two possibilities: dedicating a new Message Type (MT), e.g., MT 63, for authentication purposes or introducing a new signal component. The former approach has the advantage of not requiring hardware modifications on the receiver side, but it is likely to affect the actual system performance and may require a new certification of SBAS for the use in aviation. Furthermore, the performance of the authentication service itself may not be able to cope with stringent requirements such as those for aviation. The latter approach is an attractive solution due to higher freedom left in the design of the authentication mechanism, without degrading the SBAS operations. Indeed, it is possible to completely design the new component from the modulation to the message format. The only constraint is to minimize the effect of the new component on the current one. For instance, a viable solution – typical in the GNSS field – could be to add an in quadrature signal component. As a consequence, based on the power splitting between the two components and without increasing the transmitted power, this approach will reduce the useful received power for current users, and this in turn may increase the BER, degrading the system performance. A pictorial representation of the two solutions is given in Fig. 3.18.

The authentication messages shall carry not only the authentication of the SBAS message, but also ancillary information required in order to support autonomous users. These data shall include messages related to the key management service and information on the status of the authentication service. The key management requires a Public Key Infrastructure (PKI) used for regulate the key distribution, lifetime, renewal and, in case, revocation. Therefore, part of the authentication bandwidth shall be devoted to key management data, that in turn shall be authenticated through a signature computed with a key belonging to a higher hierarchical level. The authentication status report can be done in a similar way: if the authentication service, for any reason, is not available, the authentication message shall be substituted by a dummy message, signed using a backup key, still provided by the key management service.

An important feature of the SBAS message is its stringent time validity. Indeed, the corrections, and even more the alarms, shall be applied only during the intended time window. For this reason, the authentication mechanism shall prevent replay of the SBAS message. This can be achieved by

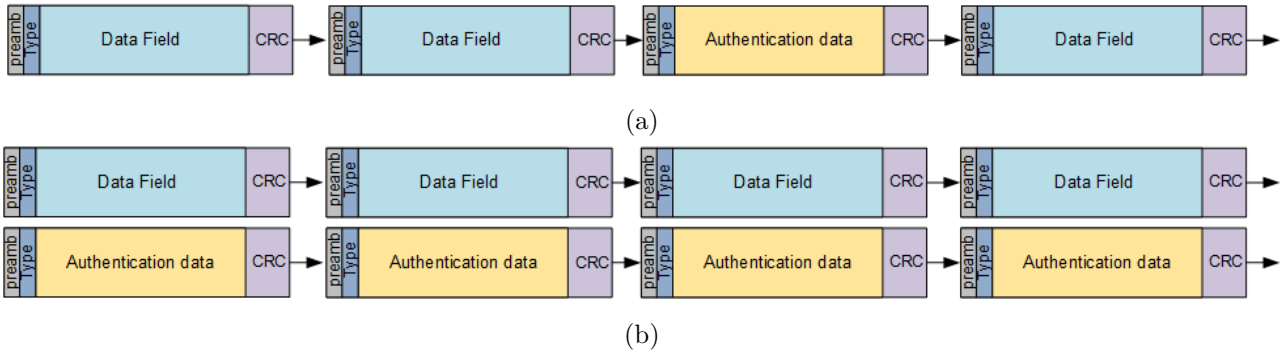


Figure 3.18: SBAS data broadcast options. The in-phase option (a) forces to insert the authentication message in the SBAS message flow, while the in-quadrature component (b) allows the broadcast of a parallel authentication data stream.

including a timestamp in the signature. This is required both for valid and dummy SBAS messages.

A generic form for the digital signature can be:

$$s = \mathbb{S}(k_{priv}, [M \parallel t]) \tag{3.91}$$

where M is the SBAS data to be authenticated and t is the timestamp of the signature transmission time. This construction can also be used in TESLA, with the difference that a MAC is computed instead of a digital signature. Moreover, in TESLA a key from the key chain shall be used and later disclosed.

For the in-quadrature solution, ideally, the modulation, the FEC encoding and the message format are designed in such a way that a signature (depending on the particular chosen scheme) can be transmitted with the same rate of SBAS messages, i.e., one signature per second. This allows to independently authenticate each SBAS message at the time of its reception, without introducing an authentication delay. This is not the case of TESLA, where a delay between the MAC and key disclosure shall be maintained for security purposes. In this case we propose that the authentication packet include the MAC of the current SBAS message, and the key used in the computation of the previous one.

In the in-phase solution, due to the bandwidth constraint, it is not possible to insert an authentication packet after every SBAS message. For this reason, we propose that each broadcast authentication message, authenticates all the SBAS messages transmitted since the previous authentication. In this case in (3.91) M becomes $M_1 \parallel \ldots \parallel M_N$, the concatenation of the not-yet-authenticated SBAS messages. This can be applied analogously to TESLA.

The in-phase solution has several drawbacks:

- the authentication delay can be considerable, e.g., a few seconds
- the authentication delay is not fixed, nor predictable, but can only be upper bounded imposing a minimum update rate to the scheduler
- the SBAS corrections should be applied after their verification, and due to the delay, this can degrade the system performance

Based on these observations we can provide some high level design recommendations:

- The introduction of a quadrature component for authentication, with an ad-hoc design, has some advantages. This is largely motivated by the expected performance, and by the fact that minimize the impact on current service.

- It is preferable to choose authentication schemes that minimize the amount of data required for authentication, to achieve fast authentication and to minimize errors.
- It is important to design an ad-hoc key management scheme that minimizes the overhead, maintaining the service open, (possibly) without aiding channels.
- Perform a risk analysis in order to optimize the scheme with respect to the user needs

3.8 Key management

Cryptography can provide many services to a communication system as confidentiality; non repudiation; integrity protection and authentication. Every system making use of cryptography shall support key management to regulate the use of cryptographic keys throughout their lifetime. In the design of a data authentication scheme the choice of parameters and the key management rules shall aim at preserving confidentiality and authenticity of the secret keys, protecting them from unauthorized use.

A *key generation* algorithm should be carefully designed in order to ensure independence between the generated instances: the leakage of a key won't compromise past or future keys. This is achieved by using fresh randomness for the generation of each new key and it is vital for the system to keep such randomness secret.

It is best practice to protect the keys by minimizing their *cryptoperiod*, the time during which a key is used before a new one is issued. A shorter cryptoperiod limits the amount of information that is protected by the same key, the time available for cryptanalytic attacks and the exposure time of the system in case of key compromise. Nevertheless, the frequency of *key update* impacts on communication overhead, since the system must broadcast key management messages. This tradeoff between bandwidth and security is a critical driver in the choice of a data authentication scheme in resource constrained systems, as security shall be maximized taking into account the limitations of the application environment.

The *key distribution* mechanism shall enable the users to receive the keys with a reasonable delay with respect to the application, and verify that they come unmodified from the intended source. A *public key infrastructure* (PKI) can be used for this purpose. The system uses a private key to sign messages, which shall be kept secret, whereas the corresponding public key can be published and used for verification.

Multiple asymmetric key pairs can be organized hierarchically: messages containing lower layer keying material shall be signed by the system with an upper layer private key; in turn, when this key needs to be updated, a key from the external layer will be used. This *key layering* structure, creates a chain of trust: each layer inherits the trust from the above layer, therefore the external layer shall be the strongest and most resistant to attacks.

This concept allows to build a *key revocation* mechanism. In case of a key compromise the system should have the possibility to prematurely end the lifetime of the current key. If the system detects such a situation, it will notify the users of the corruption and revocation of the key. For this purpose an alert message shall be broadcast and a new key be issued, and they shall be authenticated with another previously established secret key, of a higher layer in the chain of trust.

Additional services might also be offered, such as a group management mechanism to take care of different user categories, and a user revocation mechanism to allow the exclusion of subsets of users. The challenge is to provide a key management system which is able to integrate multiple services under its structure, accounting for diverse needs and service requirements.

As discussed before, revocation and renewal of asymmetric key pairs are fundamental functions in a key management system. In NMA the risk of key compromise can be reduced by introducing deterministic expiration times in order to limit the cryptoperiod, as in [49, 50]. A different problem is that of providing a method to revoke keys at random times, when there is evidence of key compromise.

Straightforward solutions exist if users are assumed to have access to the network, while this becomes a challenging problem for what concerns autonomous users, which rely on the uni-directional broadcast channel.

3.8.1 Proposal for efficient key management

In order to provide also a key management service, a hierarchical PKI can be envisioned. An example can be a three layer structure:

- CA root certificate, installed in the receiver memory, e.g., RSA with long key
- level 2, medium-term public key, used for key update/revocation, e.g., ECDSA
- level 1, short-term public key, used for NMA verification

In order to minimize the bandwidth requirement, the public keys can be preinstalled in the receiver memory. This is convenient especially for level 1 signature schemes that have short signature and long public key such as ETA, but also for using a short level 1 key size for scheme such as ECDSA reducing the level 1 cryptoperiod. In order not to leave the keys exposed for a long time, even before their entry in service, it is possible to store them encrypted. For instance, it is possible to store them in a table format that includes the key ID, the key itself and its certificate. The certificate is the signature of the public key and of some service information that bind the key to a certain context, e.g., the key ID, the SV ID and its expiration time, computed with the CA key.

When the system imposes a public key change, either due to scheduled expiration or due to revocation, it can broadcast the decryption key for the table entry corresponding to the next level 1 public key, signed with the level 2 private key currently in effect. After the demodulation and verification of the decryption key, the receiver can use it for retrieving the level 1 public key and its certificate. This solution retains the security of the long CA signatures, while minimizing the amount of data to be broadcast, transmitting only level 2 signatures.

The level 2 public key shall be updated through an aiding channel, e.g., a network link. The advantage with respect to updating directly level 1 public keys through the aiding channel, is that it is possible to reduce the level 1 cryptoperiod without forcing the receiver to frequently connect to the network.

In order to ensure service continuity the key management information must be continuously broadcast, allowing receivers to get the needed information at any time during the key validity period. This continuous broadcast of the level 2 signature also allows addressing the problem of key revocation. If each level 2 signature contains the ID of the current public key and a timestamp (e.g., the TOW) that prevents it from being replayed by an attacker, it is not possible for an attacker to intercept and exclude key revocation messages, and force the receivers to use expired (or corrupted) keys.

If we assume to use the parameters described in Section 3.5 for NMA dimensioning, recalling that the third subframe allocation was leaving 40 spare bits per subframe, it is possible to use them for the periodical transmission of level 2 signatures with low update rate.

A pictorial representation of the proposed scheme is shown in Fig. 3.19. Fig. 3.20 shows the chain of trust. The CA public key is used to verify both level 1 and 2 public key though CA certificates. The level 1 public key and its certificate are available only after the reception of the corresponding decryption key and its signature verification using the level 2 public key. The level 2 signature moreover authenticates the current level 1 key ID and a timestamp. The white blocks are stored in internal memory, while the gray blocks are broadcast by the SVs. The dashed area corresponds to encrypted storage area. It is possible to notice how this key management scheme offloads a significant part of the key management data from the broadcast channel to the internal storage.

<table>
<tr><td>Level 2 Signature</td><td>ID_N</td><td>Decryption key for ID_N</td><td>Timestamp</td></tr>
<tr><td colspan="4">Broadcast by SV</td></tr>
<tr><td colspan="4">Internally stored</td></tr>
<tr><td>Level 2 PK</td><td>ID_1</td><td>Level 1 PK #1</td><td>CA Certificate of level 1 PK #1</td></tr>
<tr><td rowspan="3">CA Certificate of level 2 PK</td><td>ID_2</td><td>Level 1 PK #2</td><td>CA Certificate of level 1 PK #2</td></tr>
<tr><td>ID_3</td><td>Level 1 PK #3</td><td>CA Certificate of level 1 PK #3</td></tr>
<tr><td>CA PK</td><td>ID_4</td><td>Level 1 PK #4</td><td>CA Certificate of level 1 PK #4</td></tr>
<tr><td colspan="2">In clear</td><td colspan="2">Encrypted</td></tr>
</table>

Figure 3.19: Proposed key management scheme. It is shown the subdivision between in clear and encrypted storage, and between information stored and broadcast.

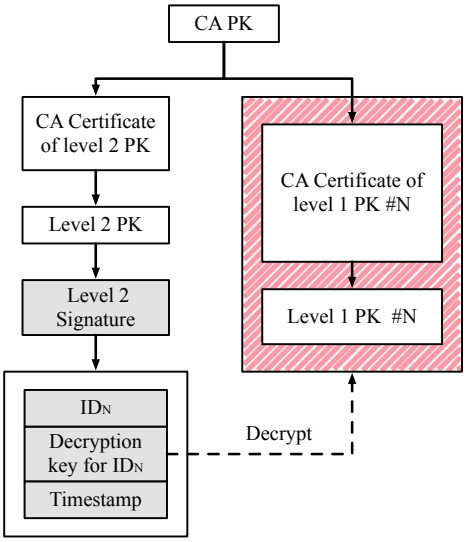


Figure 3.20: Proposed key management scheme chain of trust. The white blocks are stored in internal memory, while the gray blocks are broadcast by the SVs. The dashed area corresponds to encrypted storage area. The CA public key is used to verify both level 1 and 2 public key. The level 1 public key and its certificate are available only after the reception of the corresponding decryption key and its signature verification using the level 2 public key.

3.9 Discussion on data level authentication

Each of the schemes mentioned so far provides a different tradeoff among the design drivers highlighted in Section 3.1. A pictorial representation of this tradeoff is provided in Fig. 3.21.

In general symmetric key based schemes offer very good overall performance at the price of a poor key management scalability. This is a major issue, that can be solved only by requiring tamper resistant security modules (Fig. 3.21a, in blue). If these devices are not an option, a different approach should be considered. Asymmetric key based schemes solve this issue providing an optimal solution in terms of key management and security, but reducing the performance in all the other requirements (Fig. 3.21a, in green). One time signatures can achieve good performance in terms of computational complexity, with no other outstanding point of merit, and major drawbacks in terms of memory and communication overhead (Fig. 3.21b, in gray). Post-quantum signatures, such as code-based or Rainbow signatures, are believed to be secure against quantum computer attacks and can produce short digital signature, but requires very big public key, that might render unfeasible the Over-The-Air Rekeying (OTAR) and require more storage space in the receiver (Fig. 3.21b, in orange).

The performance of NMA schemes based on one-way chains lies in the middle, achieving more balanced performance in all the requirements. The various adaptations of TESLA to GNSS achieve good overall performance, but require trading security for communication overhead in a delicate design choice since optimality can not be achieved for both (Fig. 3.21c). Moreover, the use of TESLA requires security critical assumption on the receiver such as time synchronization and processing logic. Rather than compromising on security to achieve desirable performance, SigAm could be a promising alternative allowing gains in security and communication overhead at the cost of a longer TTFAF (Fig. 3.21d).

Table 3.3 provides the rationale for the qualitative comparison.

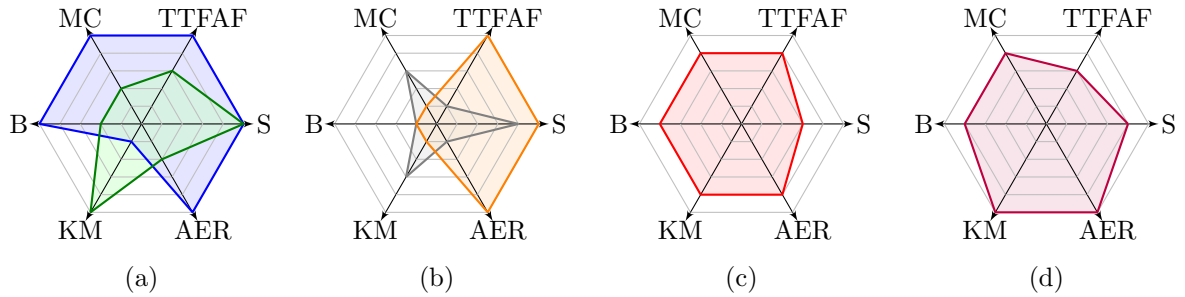


Figure 3.21: Performance comparison of different classes of data level authentication schemes. In (a) symmetric based (blue) and asymmetric based (green), in (b) one time signature (gray) and post-quantum signatures (orange), in (c) TESLA based, in (d) SigAm based. The performance are expressed in terms of: Security (S); TTFAF; Memory and Computational complexity (MC); Bandwidth requirements (B); Key Management (KM); and AER.

	Symmetric based	Asymmetric based	One time signatures	Post-quantum signatures	TESLA based	SigAm
Security (**S**)	based on well known primitives with formal security proofs	based on well known primitives with formal security proofs	less mature than traditional symmetric/asymmetric cryptographic primitives	believed to be secure against quantum computer attacks	based on non-ideal key chain, GNSS adaptation is not standardized, time synchronization is security critical	based on digital signature scheme, the protocol is not standardized, time synchronization is not needed
Authentication Error Rate (**AER**), based on the number of bits required for an authentication	MAC only	digital signature only	only digital signature, but longer	short signature	MAC + delayed key	authentication tag only
Bandwidth requirement (**B**), information that must be broadcast	MAC + symmetric key renewal	digital signature + public key renewal	digital signature + very long public key renewal	short digital signature + very long public key renewal	MAC, delayed key + signature of root key + public key renewal	digest + signature of root authentication tag + public key renewal
Scalability in terms of **Key Management** (**KM**)	less desirable situation, where all the users shares the same secret key (requires tamper resistant module)	ideal situation, in which a single key can be shared safely among all the users	the public/private key pairs can be used for a limited number of message, and their size grows with the number of message. Very big public key, difficult to perform OTAR	Very big public key, difficult to perform OTAR	similar to the asymmetric case concerning the public key for the digital signature of the root key, but it requires also the generation of the key chain	similar to the asymmetric case concerning the public key for the digital signature of the chain. The generation of the digest chain is less security critical than the key chain of TESLA
Memory and **Computational** requirements for the receiver (**MC**)	lightweight and efficient functions, short keys to be stored	intense computational requirements	lightweight functions but long public key needs to be stored	intense computational requirements, especially for generation of the signature, very big public key to be stored	intense computational requirement at the beginning of the chain (signature of the root key) and more lightweight functions for successive authentication check	intense computational requirement at the beginning of the chain (signature of the root digest) and more lightweight functions for successive authentication check
Time To First Authenticated Fix (**TTFAF**), additional information aside from the navigation message needed for the first authentication	MAC only	digital signature only	only digital signature, but longer	short digital signature only	MAC and delayed key + current root key and its signature if they are not known	digest + current root digest and its signature if they are not known. They are more probably needed than what happens in TESLA, due to the shorter chain duration

Table 3.3: Performance comparison among the different NMA candidate schemes.

Chapter 4

Authentication at Signal level

This Chapter will discuss authentication at the signal layer in both GNSS and SBAS. The results on this topic were published in [6, 7, 33].

4.1 Introduction to signal level authentication

This Chapter will discuss authentication and integrity protection at the signal layer in GNSS. Many works have investigated the possibility of achieving authentication through the use of cryptography directly at the spreading code level. In military services, such as GPS P(Y), the public spreading code is encrypted with a secret spreading code. This is referred to as Spreading Code Encryption (SCE). This offers two advantages: first, it operates as access control mechanism, because only authorized users provided with the secret keys can have access to the high precision positioning service; second, it protects users from spoofing, because an attacker needs the secret key to generate a spoofing signal. SCE is usually based on symmetric cryptographic primitives, therefore the key used for generate the encryption sequence must be kept in a tamper resistant module, to avoid key leakage.

Signal layer authentication mechanisms need to achieve an optimal tradeoff between:

- *Security:* maximizing robustness against attacks, including the choice of parameters such as: size of keys; entropy of the transmitted signal; security of algorithms; and security of key management functions such as key establishment.

- *Communications overhead:* minimizing the bandwidth requirements of key management messages, e.g., for the renewal of cryptographic keys.

- *Robustness to channel effects:* maximizing tolerance against modification of the signal induced by the channel, especially in challenging environments.

- *Scalability of key management:* suitability of the scheme for large groups of users, particularly in relation to distribution and management of cryptographic keys.

- *Requirements of the receiver:* minimizing the hardware requirements of the receiver, in terms of processing power, buffer memory and antenna gain.

- *Authentication performance:* maximizing performance in terms of probability of detection while minimizing the false alarm probability.

4.2 Review of proposals from the literature

4.2.1 Spreading code encryption

Spreading Code Encryption (SCE) relies on the use of a reserved spreading code. This is usually obtained through the modulo-2 sum of a cryptographically secure pseudo-random sequence and a PRN sequence with good auto and cross correlation characteristics. The main purpose of using SCE may be to deny access to navigation signals by unauthorized users, as in the case of military services; but it is also possible to leverage the unpredictability of the spreading sequence to ensure that the received signal is authentic and has been originated by the claimed source. SCE is a solution that is foreseen to be used in Galileo Commercial Service (CS) [86, 87].

A possible way to exploit collaboration among multiple signal components is depicted in [79], where it was proposed that the key material needed to generate the local replica of the secret spreading code for the Galileo E6 pilot channel, namely E6C, is sent on the E1B OS signal component. The same work states that in order to preserve the maximum correlation gain, cryptographic algorithms must be restricted to GF(2) arithmetics, and thus the optimal choice is some kind of stream cipher, like the Advanced Encryption Standard (AES) in counter-mode [36], but suggest the use of a candidate solution for the eSTREAM project under profile 2 that is designed as "Stream cipher for hardware applications with restricted resources such as limited storage, gate count, or power consumption". Grain-128a [88], is based on the combination of linear and non-linear feed-back shift registers, initialized with an Initialization Vector (IV) and a cryptographic key k_{SCE_i}, usually referred to as NAVSEC key, respectively. In [89] it is shown that there are 2^{96} weak key-IV pairs in Grain-128, each leading to an all-zero Linear Feedback Shift Registers (LFSR) after the initialization phase. [89] demonstrates how to distinguish such key-stream, and how to recover the initial state.

In order to allow the delayed disclosure, in [48] the use of TESLA (Section 3.2.2) is proposed. The same work proposed to use a new IV at the beginning of each sub-frame made from the concatenation of the SV_{id} and the GST of the beginning of the subframe, zero-padded:

$$\mathrm{IV}_{\mathrm{SV}_{id}} = [\mathrm{SV}_{id} \parallel \mathrm{WN} \parallel \mathrm{TOW} \parallel 0_{1\times 60}] \tag{4.1}$$

while the key k_{SCE_i} is derived from the TESLA key k_i through an intermediate variable s_i, named short chain, of length m:

$$s_i = \begin{cases} \mathcal{H}_1(k_i) & \text{if } i = m, 2m, \dots \\ \mathcal{H}(s_{i+1}) & \text{otherwise} \end{cases} \tag{4.2}$$

$$k_{SCE_i} = \mathcal{H}_2(s_i) \tag{4.3}$$

where $\mathcal{H}$, $\mathcal{H}_1$ and $\mathcal{H}_2$ are one-way functions. This is graphically explained in Fig. 4.1. Several possibilities to distribute the key k_{SCE_i} were identified: suppose to classify the user based on the level of trust and on the kind of equipment, it is possible to allow the privileged user to know the key k_{SCE_i} in advance so that they can generate a local replica of the spreading sequence and perform tracking of E6C in real time. On the other hand, a standard user can only verify the signal authenticity once the key k_{SCE_i} is publicly disclosed, by comparing the sampled version of the signal stored in a buffer with the local replica. In order to allow a user to work in real time it is possible to provide him, through a secure channel, either with a TESLA key k_i or an element of the short chain s_i, through a secure channel. This will allow the user to have real time access to the E6C signal, but at the price of introducing a potential vulnerability in the system: if for some reason the TESLA key k_j, with $j > i$, that is stored in the memory of some user is compromised, e.g., by tampering of the receiver, both the signals E1 and E6C can be counterfeit by the attacker for the time interval $[i, j]$, while if the only short chain element s_j, with $j > i$, is disclosed to users the E1 signal remains secure even after

the tampering of a receiver. The only protection against this kind of attacks is to store the key in an anti-tampering module and use a secure channel to deliver the keys.

The advantage of this layered structure is that it allows to directly distribute the NAVSEC key or the intermediate value without exposing the key of the higher layer. This means that if a NAVSEC key is compromised, neither the intermediate short chain nor the TESLA key chain are compromised and thus the complete rekeying of the system is not required.

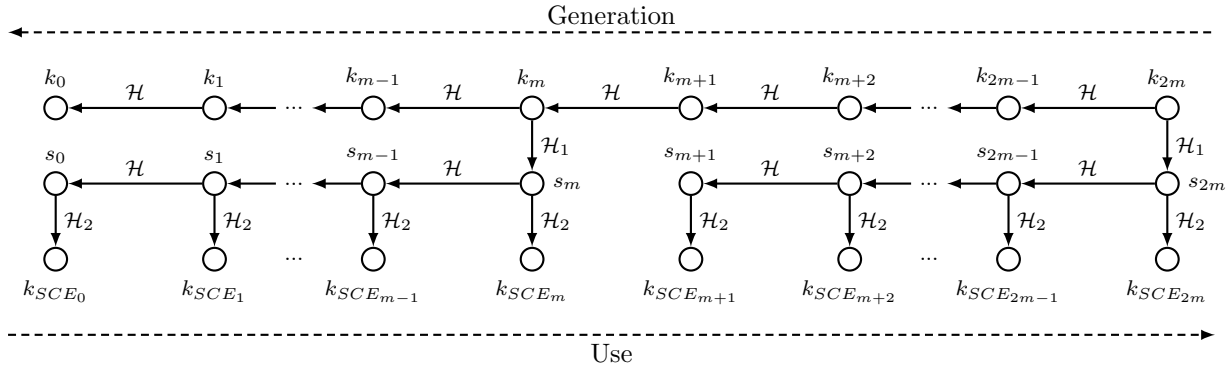


Figure 4.1: Generation of the keys used for spreading code encryption.

4.2.2 Signal watermarking

A possible solution for signal authentication is to use some form of watermarking: the transmitter includes some kind of information into the signal that is difficult for an attacker to predict or detect without the knowledge of a secret key. In this way the receiver, having some information on the watermark can check whether the received signal has it, and thus verify if the signal comes from the legitimate source.

[90] proposes an approach of this kind for GNSS. It is proposed to transmit at regular times t_m a so called *hidden marker*: a rectangular pulse of duration δ broadcast with DS-SS modulation using an unpublished spreading sequence and power spectral density chosen such that it is at least 20 dB below the thermal noise when received. The receiver is not able to verify the presence of the marker directly, but he can sample the signal and keep it in a buffer until the spreading sequence is revealed. The algorithm at the transmitter works as follows:

1. The sender, SV_{id}, generates a number $N_{id,m}$ using a secure random-number generator
2. Uses $N_{id,m}$ as seed in a cryptographically secure pseudo-random bit-sequence generator $P(N_{id,m}, j) \in \{-1, +1\}$ that outputs a sequence of bits with indices $j = \{0, 1, 2, \ldots\}$
3. From time t_m to $t_m + \delta$, the SV_{id} transmits the hidden marker $s(t)$ built as the DS-SS signal modulated by the pseudo-random sequence generated at the previous step
4. At time $t_m + \rho$, with $\rho \gg \delta$, SV_{id} broadcast a digital signature of the seed used for computing the secret spreading sequence:

$$Signature_m = Sign_{k_{priv}}([t_m \parallel \mathrm{SV}_{id} \parallel D_{id} \parallel N_{id,m}]) \tag{4.4}$$

where D_{id} are the navigation data relative to the SV_{id}

The algorithm for the receiver is:

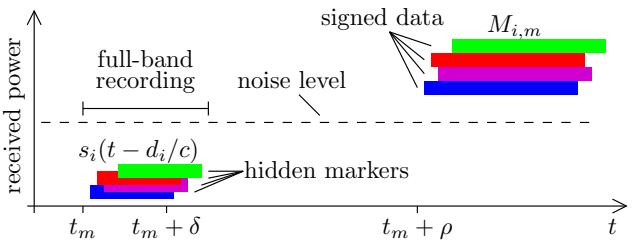


Figure 4.2: Signal authentication through watermarking as proposed by [90].

1. During a period larger that $[t_m, t_m + \delta]$ the receiver samples the frequency band $[f_c - f_s, f_c + f_s]$ with a sampling frequency of at least $4f_s$ and stores the samples in a buffer.
2. Waits for the $Signature_m$ and discards those which signature cannot be verified or whose t_m is not the one expected
3. Extracts from the $Signature_m$ the seed $N_{id,m}$, generates the secret spreading sequence and correlates it with the samples in buffer
4. Finds the position $\hat{\tau}_{id,m}$ of the largest correlation peak and the relative amplitude $\omega_{id,m}$ of any secondary peaks
5. Among the tuples $(id, \hat{\tau}_{id,m}, \omega_{id,m})$ discards all those where the ratio between the second-largest peak and the main one is above a certain threshold
6. Uses the remaining peak-positions $\hat{\tau}_{id,m}$ as pseudoranges and computes the PVT
7. Accepts the result only if the time error is smaller than the clock uncertainty and than the time delay ρ

In [90] it is also proposed a modified version of this mechanism that integrates TESLA (§3.2.2) in replacement of the digital signature, suggesting the use of the values $N_{id,m}$ as part of the one-way chain.

An attacker can perform a meaconing attack, by sampling and replaying the entire GNSS's RF spectrum after some delay. The spoofed signal will contain not only the ranging signal and the watermarking, but also the attacker's receiver noise. The received power shall be designed in such a way that in order to separate the useful signal component from noise the attacker will need a high gain directional antenna. If the attacker desires to mount selective delay attacks, he can use multiple high gain antennas. If the antenna gain is not high enough to allow separating the watermarking from the noise, the attacker could delay raw RF samples coming from each antenna and rebroadcast the resulting signal. In this case when computing the correlation with the spreading sequence the receiver will notice a presence of a secondary peak due to the legitimate signal. For this reason he must only accept signals with a secondary correlation peak that is significantly lower than the main one. Note that there are also some secondary peaks due to multipath, thus the acceptance strategy must be carefully evaluated. For instance, it is possible to use a threshold dependent on the time distance from the main peak, in order to have a good performance in terms of Receiver Operating Characteristic (ROC).

A drawback of this concept is the need of a buffer in which the receiver stores the sampled data until the disclosure of the data required to compute the secret spreading sequence. The design of the hidden marker (the time duration, the rate of the spreading code, the disclosure delay) defines the requirements for this buffer, which can easily be of a few megabytes, and may be impractical for low-end devices. Another drawback is that multiplexing another signal to the legacy signals leads to

a multiplexing loss, and so it can impact on the performance of the navigation signals and also on the performance of receivers that do not perform authentication.

4.2.3 Spread spectrum security codes

In [91] a similar approach to the watermarking for the GPS L5 signal, called Spread Spectrum Security Codes (SSSC), is proposed. The idea is to use a stream cipher to generate a spread spectrum sequence that is interleaved with the normal spreading sequence as shown in Fig. 4.3. The receiver knows when the SSSC is being received and stores the sample in a buffer until the reception of the authentication message. Once it receives the authentication message containing the secret spreading sequence, it can despread the sampled sequence and verify whether it correlates with the digitally signed version received with the message. If not, the signal must be considered as non authentic.

This architecture exhibits the same vulnerabilities and drawbacks of the Hidden Marker (Section 4.2.2) proposal. The results in [91] show that with a spoofer receive antenna gain of 20 dB the median correlation of a spoofed SSSC is 4 dB lower than the one the true SSSC. Another drawback of this scheme is that interleaving the SSSCs with the unencrypted PRN sequence can impact the performance of signal acquisition/tracking in some DLL and PLL design [92].

A revised version of SSSC targets the data component of the GPS L1C signal. In this way the receiver is able to perform a continuous tracking of the pilot component without the inconvenience of the interleaved unknown code, providing better navigation performance.

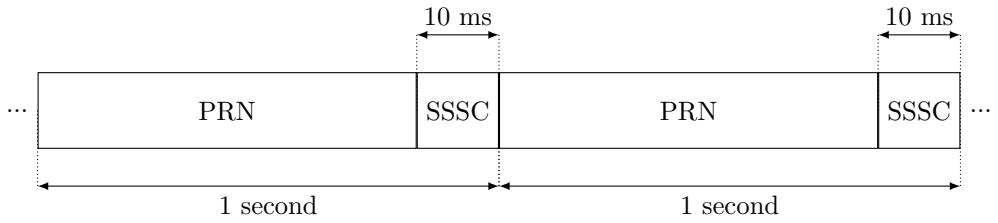


Figure 4.3: SSSC interleaved with the normal spreading code.

4.2.4 Schemes that leverage aid from Galileo CS/PRS or GPS P(Y)

In this section we introduce protocols that require interaction between the receiver and a third party server. This family of protocols requires a network link so it is not possible to use them on standalone receivers.

The main idea behind this approach is that a spoofer can easily generate the civilian navigation signal but not the encrypted signal used for military or commercial service. Several solutions have been proposed that suggest using such controlled access services in order to verify whether the received open service signal is authentic or not.

[93] proposes to use a combination between the Galileo E1 OS and E6 CS signals. The receiver can process the E1 signal as usual but it does not have access to the NAVSEC key for CS. The receiver samples the E6 band and transmits the samples to a remote server that has access to the CS service and can compute a secure PVT solution that will be checked against the one computed at the receiver.

[94] makes a similar proposal that applies to GPS. The idea is to keep a reference receiver in a secure location and correlate the sampled P(Y) signal with the one that is received by an insecure receiver. If the correlation is large enough the signal received by the second receiver is considered authentic, otherwise an alert is issued. A benefit of this solution is that the system does not require getting access to restricted service, and moreover was demonstrated that this approach also works with

narrowband C/A receivers with a RF front-end bandwidth of approximately 2 MHz, hence expensive wideband receivers are not needed.

These systems are mainly vulnerable to the following threats:

- SCER attack: as we will see in Section 4.4, the attacker can try to estimate the secret W code on the fly and use this estimated version for his replica. Depending on the capability of the attacker, like the C/N_0 and the estimation strategy used, he can correctly guess the secret chips with a certain probability.

- Multiple spoofer devices: if the attacker can use two spoofer devices, one close to the victim receiver and another close to the reference receiver, the received P(Y) code will correctly correlate even if it is not authentic. In this case the system is not giving any defense.

- Meaconing attack: in order to compute the correlation between the two receivers the samples must be aligned, it is proposed to map the inter-receiver time using the C/A code start/stop times. If an attacker can sample the entire authentic signal (open + military service) with a high end device with large bandwidth frontend, the resulting signal will correctly correlate as the original one.

A slightly different approach is presented in [95] that proposes to leave only the job of sampling the signals to the receiver and send this to a server that has access to the PRS and will return the computed PVT solution to the user. In order to reduce the amount of data that must be exchanged the navigation data will not be obtained from the signal itself because they are already known to the server, allowing the use of a short signal snapshot.

4.2.5 Physical layer authentication

Physical layer authentication is a class of authentication mechanisms that exploits some physical characteristics that are difficult to be falsified. These features can be for instance some physical non-ideality of the devices or the channel response. A way to exploit the feature related to the devices is the device fingerprinting, already discussed in Section 3.4. In [77] fingerprint was used to identify the receiver by a third party. If the receiver itself wants to authenticate the source of the received ranging signal, it should fingerprint the SV itself. Features commonly used as fingerprint are the local oscillator, non-ideality in the off-on transient, modulation errors, and the non-linearity of the power amplifier [96, 97].

Beside the transmitter characterization, it is possible to evaluate the channel response and compare the actual estimation with a previous known estimation. The basic idea in which a transmitter and a receiver, that share a secret key, use a statistical hypothesis testing was introduced by [98], and the idea was improved in [99] taking into account also a noisy channel. In [100] a strategy which takes into account also the possibility that the attacker tries to influence the channel estimation of the receiver in a Multiple-Input and Multiple-Output (MIMO) environment was introduced. The scheme is based on two phases: in the first the receiver builds its reference channel estimation and in the second it performs the detection in real-time, where the channel estimation of the current packet is compared to the previously authenticated one.

In the GNSS context the fingerprinting of the transmitter is difficult due its peculiarities: the transmission is continuous, so it is not possible to evaluate the transient, and due to the high amount of noise it is difficult to evaluate characteristics such as modulation error. Also the channel estimation is difficult to be used in the GNSS environment, especially in the open field this solution because the channel estimation seen by an attacker on ground is typically very close to the channel seen by the receiver, and so the attacker can mimic the latter very well. This may be not true in an urban environment in which the estimation made by the attacker can be different that the receiver's one, but

in this case the problem can be the short coherence time and the device mobility. For these reasons, the investigations of these techniques in the GNSS context is left for future work, and the use of channel estimation has instead been applied to the IoT context, which will be discussed in Chapter 5.

4.3 Analysis of state-of-the-art techniques

In general terms encryption does not provide authentication, e.g., One Time Pad (OTP) is the optimal encryption scheme providing perfect confidentiality but does not protect against data modification. The reason why it is generally assumed that SCE, either in its direct form or as proposed in SSSC or hidden markers, offer a form of signal authentication is that, due to the long spreading sequence used, the signal reaches the receiver antenna with a power spectral density that is lower than the noise floor. Since the spreading code is unpredictable, it can not be used to obtain the processing gain and recover the original signal. In order to recover the secret spreading code a high gain antenna, e.g., big dish antenna, is needed, making intractable for an attacker to obtain a noiseless replica of the spreading code used. If a naive noisy estimation is performed by the attacker and it is used to generate a spoofing signal, this will, in general, correlate poorly with the local replica of the receiver that has access to the cryptographic material used to generate the secret spreading code. This kind of security can be seen as a traditional computational security paradigm, where the system is secure due to the limited resources available to the attacker. In this case, instead of the computational power, the resources are represented by the number of antennas and their gain. Indeed, if the attacker has many high gain dish antennas, each pointing to a SV, it can extract a sufficiently clean signal to be used to spoof the target device. On the other hand, this high gain antennas are expensive and non portable, thus it is intractable for non sophisticated attackers to have access to a clean signal estimation, guaranteeing a certain security level to the scheme.

However, signal authentication can be achieved through signal encryption only when additional constraints are used to restrict the attacker freedom. For instance, the receiver shall be tamper resistant, avoiding the possibility that the attacker have access to the antenna connector or influence the local oscillator. Otherwise, a trivial attack is as easy as inserting some extra wiring between the antenna and the receiver. Clearly this attack has a limited utility, but allows the attacker to influence the PVT computation without failing the signal layer authentication checks. This toy example shows that it is not correct to assume that a single signal feature or a single technique is able to provide an authenticated PVT.

SigAm, the proposed NMA scheme (Section 3.5), can be used as alternative to TESLA in the context of signal layer authentication. For instance, in [79], the authentication tags $H_i(n)$ of SigAm can be used in place of the TESLA key k_n and, since SigAm is more bandwidth efficient with respect to TESLA, the NAVSEC key can be easily fitted in the subframe. SigAm can bring all the security benefit discussed above, such as shorter multiple chains and lack of time synchronization requirement, with the drawback that it is not possible to distribute keys for real time generation of the spreading codes, thus the trusted receiver shall be re-keyed more frequently through the aiding channel.

An even more interesting combination between signal and data level techniques is the use of SigAm authentication tag as seed for the generation of the secret spreading code used for the watermarking or in SSSC. Even in this case, as SigAm is more bandwidth efficient than TESLA, it allows achieving a shorter TBA, but the main addition to the original watermarking concept in [90] is that the RNG seed depends on both the signing key and the message instead of an independent key. This adds a link between the authentication at the navigation message and at the one at signal layer in an efficient way. As happens with the TMBOC, the inserted watermark can also have a higher chipping rate than the normal one, thus increasing the required antenna gain for a successful attack.

It is noteworthy that this solution allows scalability: low-end devices not interested in authentication can process the ranging signal only, while receivers that require authentication can demodulate

the navigation message authentication and can check the presence of the watermark with the desired authentication rate. Thus, it allows the receiver manufacturer to design the receiver based on the user requirements, without forcing them to implement unnecessary verification features or to perform it more often than needed.

4.4 Security code estimation and replay attack

The SCER attack was introduced in [30] as a possible threat for all the schemes that use cryptographic protection of the GNSS signal.

The goal of a SCER attack [30, 6] is to estimate a legitimate signal in order to generate a spoofed signal with minimal delay (if any). It is important to note that the estimation need not be perfect, only good enough so that the reproduced signal is indistinguishable from the authentic one, when corrupted by channel and noise at the receiver.

This type of attack represents a general threat that can be carried out irrespective of the particular cryptographic schemes employed in the signal (be it data layer schemes: e.g., NMA; or signal layer schemes: e.g., SCE, SSSC). The victim will receive both the authentic and spoofed signals below the noise floor; only after correlation, and by exploiting the processing gain, does the SNR become sufficiently good to reliably process the signal.

If the SNR of the attacker is significantly lower than the victim's, he's estimation error will introduce noticeable effects on the spoofed signal as seen by the victim receiver. If, on the opposite, the attacker is in better conditions than the victim, such effects will be hidden by the receiver noise.

In order to maximize the information obtained from the received signals, the detection statistic is taken in principle before the correlation. Indeed, the correlation process increases the SNR thus reducing the noise contribution, and hence the detection capability.

The attacker's estimation will improve over time with the accumulated energy in the symbol (data level) or chip (signal level). Thus, after an initial transient, the attacker estimation becomes reliable and the difference between the legitimate and estimated signals tends to vanish, as we will see in the following.

4.4.1 System model

Similarly to [30], we assume a GNSS signal equipped with some *security code* such as SSSC [91] or NMA [32], and some public spreading code and data, that is

$$x_0(t) = w_n c_m \cos(2\pi f_0 t + \varphi_0) \quad , \quad t \in [nT_w, (n+1)T_w] \cap [mT_c, (m+1)T_c], \tag{4.5}$$

where:

- w_n is the security code (T_w being its symbol period) that is unknown to the attacker,
- c_m is the combination of navigation data, PRN codes and every other transmitted data that is publicly known or predictable, with symbol interval T_c,
- f_0 and φ_0 are the carrier frequency and phase, respectively.

We observe that not only does this model encompass all the techniques in which signal authentication is provided by means of spreading codes such as the P(Y) [11], Signal Authentication Sequence (SAS) [92], supersonic codes [101], etc., but also those in which the navigation message is protected by some symmetric MAC [48, 49, 79] or by a digital signature scheme [32], with $\{w_n\}$ representing the MAC or the signature that is unknown to the attacker prior to transmission.

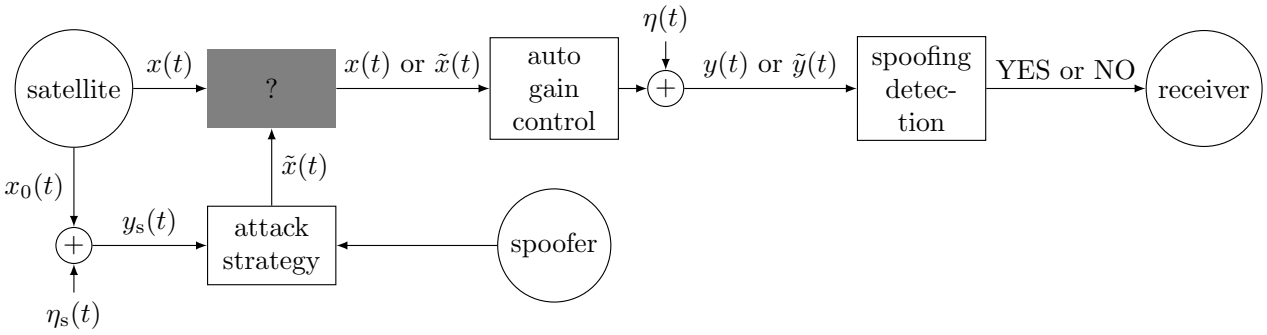


Figure 4.4: Block-diagram of the system model for SCER.

As shown in Fig. 4.4, we consider a spoofing attacker that upon receiving the signal

$$y_{\mathrm{s}}(t) = x_0(t) + \eta_{\mathrm{s}}(t) \tag{4.6}$$

where $\eta(t)$ is the additive noise (assumed white and Gaussian) at the spoofer side, aims at forging the victim's position and timing fix, by carrying out either of the following attacks, or both

1. replacing the true navigation data in $\{c_m\}$ by a forged or modified $\{\tilde{c}_m\}$
2. selectively delaying the signal $x_0(t)$ by some T_{d}

and having the victim accept the illegitimately modified signal $\tilde{x}(t)$ as if it were the following authentic signal

$$\tilde{x}(t) = w_n \tilde{c}_m \cos(2\pi f_0(t-T_{\mathrm{d}})+\varphi_0) \quad , \quad t \in [nT_w+T_{\mathrm{d}}, (n+1)T_w+T_{\mathrm{d}}] \cap [mT_c+T_{\mathrm{d}}, (m+1)T_c+T_{\mathrm{d}}] \, . \tag{4.7}$$

Since the actual value of w_n is not known to the attacker, in carrying out the above attack he must transmit some signal $\tilde{x}(t)$ that resembles $x(t)$ as much as possible, and that is based on what is publicly known (or predictable) and what he has so far observed from his received signal y_{s}. In particular, at any instant t, he can build the value $\tilde{x}(t)$ by making use of past and current samples $\{y_{\mathrm{s}}(u), u \leq t\}$. The choice of how to make $\tilde{x}(t)$ depend on $\{y_{\mathrm{s}}(u), u \leq t\}$ represents the attacker strategy.

On the contrary, the receiver aims to discriminate the forged or artificially delayed signal from the authentic one. To this purpose, he applies a binary hypothesis testing to the received signal in an interval of duration T_{f}, given by

$$y(t) = \frac{x(t)}{\sqrt{\mathrm{E}\left[x^2(t)\right]}} + \eta(t) \quad \text{or} \quad \tilde{y}(t) = \frac{\tilde{x}(t)}{\sqrt{\mathrm{E}\left[\tilde{x}^2(t)\right]}} + \eta(t) \quad , \quad t \in [t_0, t_0 + T_{\mathrm{f}}] \tag{4.8}$$

where the effect of the input scaling performed by the automatic gain control is explicitly shown and $\eta(t)$ denotes additive white Gaussian noise. In doing so, the receiver aims at keeping both the probability of not detecting a forged signal (*type II* error, or *missed detection*, probability p_{md}) and that of erroneously marking a legitimate signal as spoofed (*type I* error, or *false alarm*, probability p_{fa}) below an acceptable level. Clearly, a tradeoff must be sought between minimizing p_{md} and p_{fa}, as, for instance, rejecting all messages as spoofed allows to have $p_{\mathrm{md}} = 0$ at the cost of having $p_{\mathrm{fa}} = 1$. However, from the theory of binary hypothesis testing, we know that the optimal test (that is the one that yields the minimum p_{md} for any given constraint on p_{fa}, and *vice versa*) is given by the Likelihood Ratio Test (LRT), also known as *Neyman-Pearson criterion* [102, §3.3][98] if both the statistics of the legitimate and spoofed signal are known (that is if the victim is aware of the particular strategy adopted by the spoofer). Thus, given a T_{s}-sampled version $\boldsymbol{y} = [y(t_0), y(t_0 + T_{\mathrm{s}}), y(t_0 + 2T_{\mathrm{s}}), \ldots, y(t_0 + T_{\mathrm{f}})]$ of

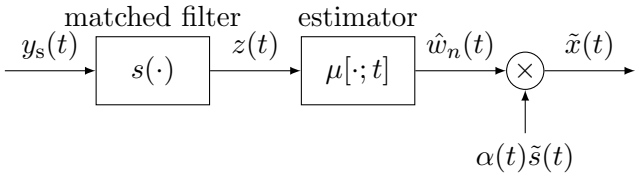


Figure 4.5: Block diagram representation of the SCER attack. The signal $\tilde{s}(t) = \tilde{c}_m \cos(2\pi f_0(t - T_d) + \varphi_0)$ represents the signal that the attacker wants to forge, while $\alpha(t)$ is a spoofer-controlled (possibly time-varying) gain.

the received signal in the observation window, and denoting by $p_{\boldsymbol{y}}(\cdot)$ and $p_{\tilde{\boldsymbol{y}}(\cdot)}$ the pdf of the authentic and illegitimate signal, respectively, the detector deems it legitimate if

$$L(\boldsymbol{y}) = \ln \frac{p_{\boldsymbol{y}}(\boldsymbol{y})}{p_{\tilde{\boldsymbol{y}}}(\boldsymbol{y})} \geq \vartheta \tag{4.9}$$

for some particular value of the threshold ϑ, the choice of which governs the tradeoff between false alarm and missed detection probabilities. In fact, p_{fa} increases with ϑ, while p_{md} decreases as ϑ increases.

On the other hand, it is reasonable to assume that the victim is unaware of the particular attack strategy employed by the spoofer. Hence, only the statistics $p_{\boldsymbol{y}}(\cdot)$ of the authentic signal are known at the receiver, while a whole set of possible distributions $\left\{p_{\tilde{\boldsymbol{y}}(\cdot)}\right\}$ must be considered for the spoofed signal. In this case the LRT no longer applies, and a quite effective solution is to use the Generalized LRT (GLRT) [102, §6.4], (although its optimality has only been proven in particular cases [103]), that is done by deeming $y(t)$ legitimate if

$$G(\boldsymbol{y}) = \ln \frac{p_{\boldsymbol{y}}(\boldsymbol{y})}{\max_{\tilde{\boldsymbol{y}}} p_{\tilde{\boldsymbol{y}}}(\boldsymbol{y})} \geq \vartheta \tag{4.10}$$

where the maximum in the denominator of $G(\boldsymbol{y})$ is taken over all possible attack statistics.

In the following sections we detail the possible attacks and derive detection strategies, aiming to optimize both sides.

4.4.2 Attack strategies

In order to make the spoofed signal $\tilde{x}(t)$ mimic the authentic $x(t)$ as much as possible, the attacker should guess the secret code w_n. This can be done at the data level, by attempting at breaking the cryptographic scheme that generates the secure code. However, the SCER attack takes a different route, operating at the physical layer. By considering w_n as a sequence of i.i.d. random (binary[1]) symbols, it aims at continuously estimating a likely value for each w_n from the observation of the received signal in the corresponding symbol interval. Such an attack represents a very general threat, that is completely agnostic with respect to the particular cryptographic mechanism employed by the transmitter to authenticate the data.

In general, deriving an efficient estimate $\hat{w}_n(t)$ requires processing the incoming signal $y_s(t)$ with non linear, non memoryless, and time-varying transformations, the optimization of which is a difficult problem. A clever idea in [30] is to split the estimation problem into the (possibly suboptimal) scheme illustrated in Fig. 4.5, as the cascade of a matched filter (linear, with memory) and a nonlinear, instantaneous, possibly time-varying estimator.

[1] although the same technique can be easily adapted to higher symbol cardinalities

The matched filter has input-output relationship

$$z(t) = \frac{1}{2(\bar{t} - nT_w)} \int_{nT_w}^{\bar{t}} y_{\mathrm{s}}(u)\, s(u)\, du \quad , \quad t \in [nT_w + T_{\mathrm{d}}, (n+1)T_w + T_{\mathrm{d}}] \tag{4.11}$$

with

$$\bar{t} = \min\{t, (n+1)T_w\} \quad , \quad s(u) = c_m \cos(2\pi f_0 u + \varphi_0) \tag{4.12}$$

and collects the useful contribution of the received signal up to $\bar{t}$, at the same time reducing the effect of noise. Thus, conditioning on the true value w_n of the secret sample, $z(t)$ is Gaussian distributed, with mean w_n and variance that decreases in an inverse proportional fashion to $\bar{t} - nT_w$. Observe that, for $T_{\mathrm{d}} < T_w$, increasing T_{d} allows to reduce the variance of $z(t)$. This is consistent with the intuitive notion that the attacker can leverage the artificially created delay to observe a larger portion of the signal and get a more reliable guess of each secret code symbol. However, it should be noted that T_{d} can not be increased at will, as a large delay will alert the receiver[2].

The instantaneous estimator can be designed as a deterministic real function of a single real variable, mapping $z(t)$ to the estimate $\hat{w}_n(t)$. In [30] three cases are considered for the estimator function $\mu[\cdot; t]$, two time-invariant, the ML and Maximum *A Posteriori* (MAP) estimators, and one time-varying, the Minimum Mean Square Error (MMSE) estimator:

$$\mu_{\mathrm{ML}}[z] = z \quad , \quad \mu_{\mathrm{MAP}}[z] = \mathrm{sgn}(z) \quad , \quad \mu_{\mathrm{MMSE}}[z; t] = \tanh\left(\frac{z}{\sigma_z^2(t)}\right). \tag{4.13}$$

In fact, although the MAP estimator is optimal for the purpose of minimizing the estimate error probability $\mathrm{P}[\hat{w}_n(t) \neq w_n]$, nothing can be stated about optimality in providing a signal $\tilde{x}(t)$ that is hardly distinguishable from $x(t)$ (e.g., close in terms of some statistical distance). Indeed, it was shown in [30] that, in terms of probability of missed detection by the victim, each of the three estimators considered there, outperforms the remaining two in some cases, depending on the system parameters. As regards the scaling factor α, in [29] it was considered constant and arbitrarily chosen.

With the aim of deriving an optimal estimator, and hence a more consistent worst-case scenario from the point of view of the detector, we consider a more general

$$\mu[z; t] = \begin{cases} \rho(t) z & , \ |z| < 1/\rho(t) \\ \mathrm{sgn}(z) & , \ |z| \geq 1/\rho(t) \end{cases} \tag{4.14}$$

and choose $\rho(t)$ jointly with $\alpha(t)$ in order to minimize the Kullback-Leibler (K-L) divergence[3] between the authentic and the forged signal. Observe that our choice clearly generalizes that in [30], as it can be easily shown that

$$\lim_{\rho \to 0} \frac{1}{\rho} \mu[z; t] = \mu_{\mathrm{ML}}[z] \quad , \quad \lim_{\rho \to \infty} \mu[z] = \mu_{\mathrm{MAP}}[z] \quad , \quad \rho(t) = \frac{1}{\sigma_z^2(t)} \Rightarrow \mu[z; t] \simeq \mu_{\mathrm{MMSE}}[z; t]\,. \tag{4.15}$$

The different estimation strategies are shown in Fig. 4.6, where Fig. 4.6a shows the detection strategies from (4.13), and Fig. 4.6b shows some possible realizations chosen in the class described by (4.14).

This choice is motivated by the fact that the K-L divergence is a well established measure of statistical distance and that it allows to set a lower bound to the ROC of any binary hypothesis testing scheme. Although such bounds turn out to be rather loose for Gaussian variables, we shall see in Section 4.4.4 that attacks chosen according to this criterion actually yield higher values of p_{md} and p_{fa}.

[2]Roughly speaking, a GNSS receiver that has a relative clock stability of δ has to be blinded for a time interval T_{d}/δ in order to prevent it from reliably detecting a spurious delay T_{d}.

[3]The K-L divergence between two probability density functions p and q is defined as $D(p||q) = \int p(u) \log_2 \frac{p(u)}{q(u)}$

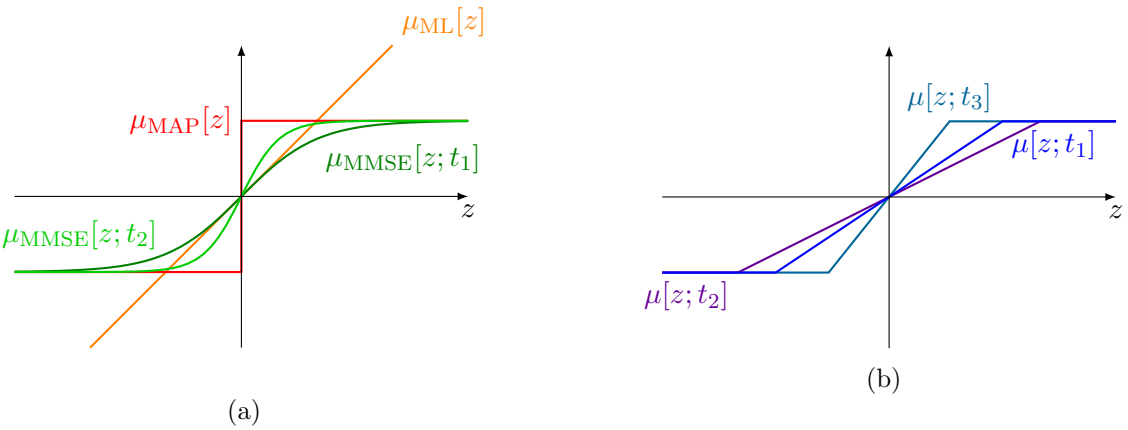


Figure 4.6: SCER detection strategies: Humphreys' in (a) and ours proposed in (b).

Ideally we would aim to minimize the K-L divergence $D(p_{\boldsymbol{y}}||p_{\tilde{\boldsymbol{y}}})$ between the random vectors representing the sampled received signal over all possible time-varying strategies $(\rho(t), \alpha(t))$. However, such optimization is exceedingly complex for typical values of T_w and T_s. Therefore, we consider two intermediate choices:

- a time-invariant scheme obtained from the minimization of the divergence between vectors

$$(\rho^\star, \alpha^\star) = \arg\min_{\rho,\alpha} D(p_{\boldsymbol{y}}||p_{\tilde{\boldsymbol{y}}}) \tag{4.16}$$

- a time-varying minimization of the divergence between corresponding pairs of single samples

$$(\rho^\star(t), \alpha^\star(t)) = \arg\min_{\rho,\alpha} D(p_{y(t)}||p_{\tilde{y}(t)}) \tag{4.17}$$

4.4.3 Detection scheme

In order to derive the explicit expression of the LRT detection scheme, we consider that, once the secure code w_n is known (as is assumed for the legitimate receiver), the received authentic signal $y(t)$ is Gaussian distributed with mean $x(t)/\sqrt{\mathrm{E}\left[x^2(t)\right]}$ and independent samples with the same variance σ_η^2 as the additive noise. As such, the authentic $\boldsymbol{y}$ is a Gaussian vector with mean vector $\boldsymbol{m} = [x(t_0), \cdot, x(t_0+T_\mathrm{f})]/\sqrt{\mathrm{E}\left[x^2(t)\right]}$, and covariance matrix $\boldsymbol{K} = \sigma_\eta^2 \boldsymbol{I}$, with $\boldsymbol{I}$ denoting the identity matrix.

The above derivation can not be repeated for the spoofed signal, and Gaussianity cannot be straightforwardly assumed due to the estimator nonlinearity. However, we approximate it with a Gaussian distribution, as well, under the justification that the AWGN noise at both receivers (the spoofer and the victim) makes up for a large component of the signal statistics. Moreover, in the spoofed case samples are correlated, due to the correlation between estimates $\hat{w}_n(t)$ at different t of the same symbol w_n, and the covariance matrix $\tilde{\boldsymbol{K}}$ is no longer diagonal. Yet, since the attack strategy estimates each w_n independently, $\tilde{\boldsymbol{K}}$ is block diagonal, with each block corresponding to the samples in the same symbol interval of the secret code.

For Gaussian vectors $\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{m}, \boldsymbol{K})$, $\tilde{\boldsymbol{y}} \sim \mathcal{N}(\tilde{\boldsymbol{m}}, \tilde{\boldsymbol{K}})$ the LRT criterion (4.9) can be stated as

$$L(\boldsymbol{y}) = (\boldsymbol{y}-\boldsymbol{m})^T \boldsymbol{K}^{-1}(\boldsymbol{y}-\boldsymbol{m}) - (\boldsymbol{y}-\tilde{\boldsymbol{m}})^T \tilde{\boldsymbol{K}}^{-1}(\boldsymbol{y}-\tilde{\boldsymbol{m}}) \tag{4.18}$$

In [30] the above expression is simplified, for the sake of a lower computational complexity, by neglecting the fact that $\boldsymbol{K} \neq \tilde{\boldsymbol{K}}$, and hence use is made only of the different means $\boldsymbol{m} \neq \tilde{\boldsymbol{m}}$ in discriminating the two signals. It should be noted that such simplification does not remove the need for the victim

to know the attack strategy, as $\tilde{\boldsymbol{m}}$ depends on it. We shall also see in Section 4.4.4 that such simplification leads to an important performance loss in terms of the detector ROC, especially against time-varying attacks.

On the other hand, it is particularly interesting to consider the supposedly more realistic case, in which the detector has no information on the attack strategy, and hence of the particular values of (ρ, α) chosen by the spoofer. Hence, he must revert to the GLRT (4.10), which, again under the hypothesis of approximate Gaussianity, yields

$$G(\boldsymbol{y}) = (\boldsymbol{y} - \boldsymbol{m})^T \boldsymbol{K}^{-1} (\boldsymbol{y} - \boldsymbol{m}) - \min_{\rho,\alpha} [(\boldsymbol{y} - \tilde{\boldsymbol{m}})^T \tilde{\boldsymbol{K}}^{-1} (\boldsymbol{y} - \tilde{\boldsymbol{m}})] \tag{4.19}$$

4.4.4 Numerical results

In this section we present simulation results to illustrate the performance of the proposed detection schemes in the presence of different attacks.

Fig. 4.7a shows the performance of both the proposed LRT and the simplified version of [30] against attacks with no delay ($T_\mathrm{d} = 0$). Although the signal observation period T_f has been chosen rather small and hence p_fa and p_md have impractically high values (to ease visualization of results), it is clearly seen that:

- the proposed attack, with optimal choice of $(\rho^\star, \alpha^\star)$, is most effective in increasing the ROC of both detection schemes;
- the proposed LRT detection scheme performs better than the approximate one against all the attacks considered

When the spoofer is allowed to inject a large delay into the signal, as shown in Fig. 4.7b, his estimate of the secret code can be very reliable, and it is very hard for the victim to distinguish an authentic signal from a spoofed one. Therefore, the values of p_fa and p_md further increase with respect to Fig. 4.7a. Moreover, the advantage of LRT over the approximate version nearly vanishes, as does the gain of the optimal attack over the MMSE and MAP.

In Fig. 4.7c, we show the detection performance with time-varying attacks, where coefficients ($\alpha^\star(t)$ for all attacks, and $\rho^\star(t)$ for the proposed attack) are chosen to minimize the K-L divergence at the single sample level. The system parameters are as in Fig. 4.7a, and again no delay is allowed. It is clear from the plots that in this case the performance gap between the proposed LRT and the approximate one has largely increased. This is due to the fact that the approximate LRT only distinguishes the authentic signal from the spoofed one based on their time-varying statistical mean, which is a first order statistical parameter that can be easily mimicked with a time-varying coefficient. On the contrary, the proposed LRT makes use of cross correlations between samples and hence it can better distinguish an attack that does not imitate the joint distribution of samples, only their marginals.

Fig. 4.7d shows the performance of the GLRT detection scheme. Although the error probability ROC increases in general with respect to the LRT case, it is seen that such a solution can still work in detecting spoofing, without any knowledge of the spoofer strategy.

In order to observe practically meaningful values for p_fa and p_md, one has to consider longer signal observation period T_f. Clearly, by observing more samples before taking his decision, the receiver will base it on more information, and the decision will be more accurate, but this requires to buffer and process more data. Furthermore, this leads to a longer TTA. Thus, the observation period must be chosen as a trade off between the computational resources of the device, the desired TTA and the desired performance in terms of ROC.

In the following we will compute the detection metric over 400 secret chips, thus using $T_\mathrm{f} = 0.8\,\mathrm{ms}$ for GPS P(Y) scenario and $T_\mathrm{f} = 1.6\,\mathrm{s}$ for Galileo E1 OS scenario. For the LRT detection, Fig. 4.8 shows the performance achieved on a SCE scheme with system parameters compliant with GPS P(Y) signal

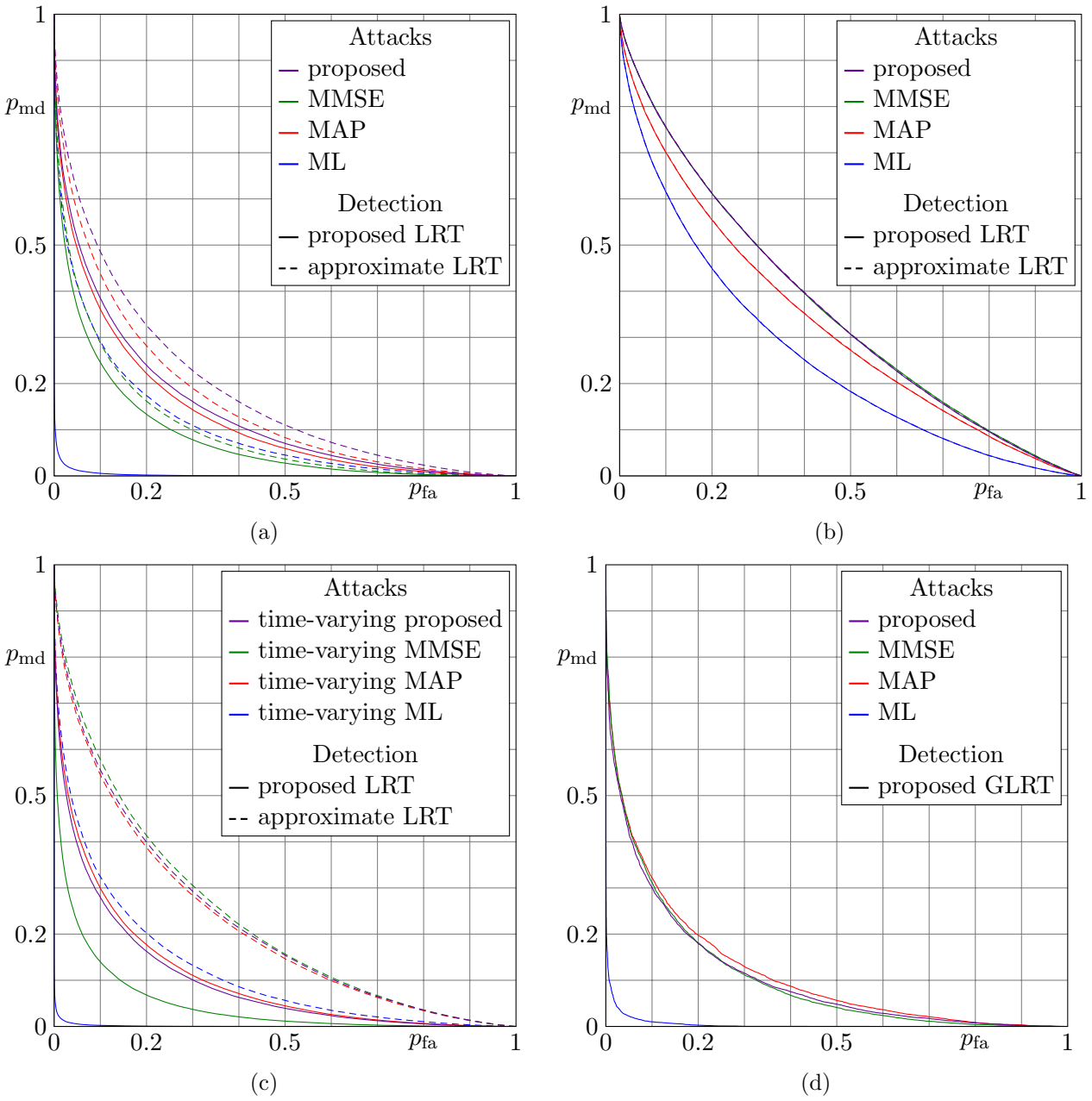


Figure 4.7: (a) – (c): ROCs for the LRT detection method proposed in this paper and the one in [30], with the optimal SCER attack proposed here and those considered in [30]. System parameters for all three plots are: $(C/N_0)_{\mathrm{att}} = 54\,\mathrm{dB}$, $(C/N_0)_{\mathrm{rec}} = 48\,\mathrm{dB}$, $T_c = 0.98\,\mu\mathrm{s}$, $T_w = 4\,\mathrm{ms}$, $T_{\mathrm{s}} = 18\,\mathrm{ns}$, $T_{\mathrm{f}} = 160\,\mathrm{ms}$. In particular, the values for T_w, T_c are consistent with NMA on Galileo E1 signal. In (a) and (c), $T_{\mathrm{d}} = 0$, while in (b), $T_{\mathrm{d}} = 4.9\,\mu\mathrm{s}$. In (c), time-varying attack strategies are considered. (d): ROCs for the GLRT detection method proposed, system parameters and attack strategies are as in (a).

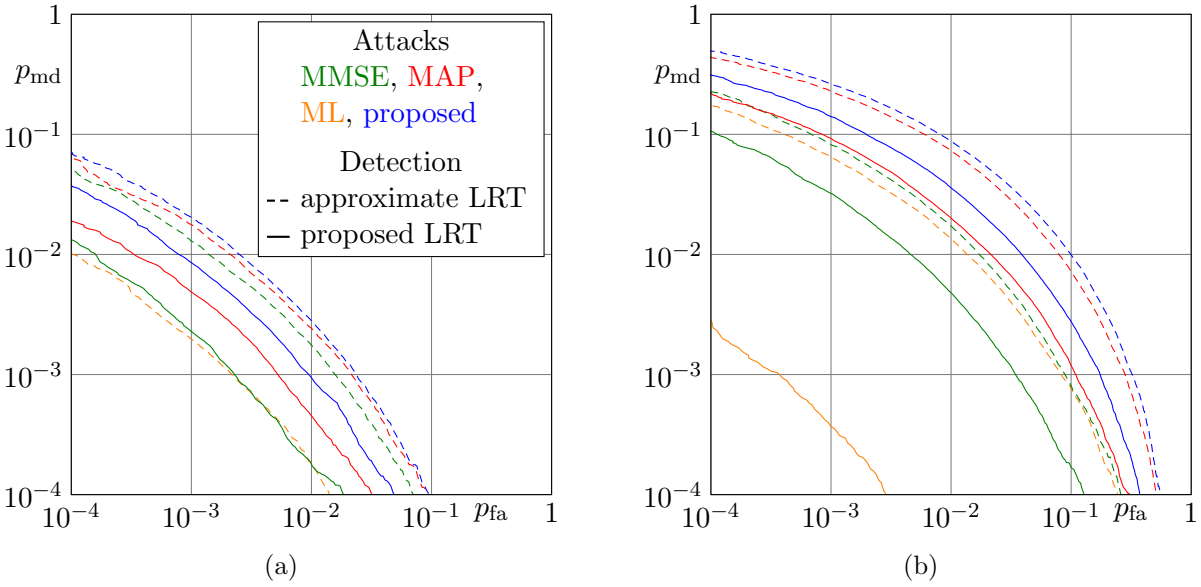


Figure 4.8: ROCs for the LRT detection method proposed in this paper and the one in [30], with the optimal SCER attack proposed here and those considered in [30]. System parameters are consistent with SCE on GPS P(Y): $(C/N_0)_{\text{rec}} = 48\,\text{dB}$, $T_{\text{d}} = 0\mu\text{s}$, $T_w = 2\,\mu\text{s}$, $T_{\text{f}} = 0.8\,\text{ms}$. In (a) $(C/N_0)_{\text{att}} = 51\,\text{dB}$, while in (b) $(C/N_0)_{\text{att}} = 54\,\text{dB}$.

characteristics and a C/N0 advantage of 3 dB and 6 dB, respectively, for the attacker. On the other hand, Fig. 4.9 shows the performance achieved on a NMA scheme with system parameters compliant with E1 OS. Here, the advantage of both the proposed attack strategy, and detection scheme is again quite self-evident.

Moving to the GLRT detection, Fig. 4.10 shows the ROCs for the GPS P(Y) scenario with a C/N_0 advantage of 3 dB for the attacker. Fig. 4.11 shows how the ROC varies as function of the C/N_0 advantage for the attacker and delay. We can see that the ROC moves quickly towards the gray dotted line that represent the trivial limit case in which the decision is taken without looking at the signal, but tossing a biased coin.

The same analysis is generalized in Fig. 4.12 and Fig. 4.13 for the LRT detection respectively for the GPS P(Y) and the Galileo E1 OS, for a fixed $p_{\text{fa}} = 10^{-2}$. We see that if the attacker has a significant advantage over the user the detection of the attack becomes way more difficult, especially for a significant delay advantage. In order to protect against this the receiver should have a small uncertainty on the clock, so that for the attacker is not feasible to obtain a big advantage.

Another aspect that arise form these figures is the fact that the gain of the proposed LRT scheme is maximum for the zero delay attack and decrease when the delay increase. This is due to the fact that increasing the delay reduce the difference of the covariance matrix from the approximate value used in [30].

4.4.5 Results on realistic signals

In order to assess the effectiveness of the SCER attach, an experiment was devised involving the use of real signals. Two scenarios were investigated:

1. *Static scenario:* in this case the signal was acquired through an omni-directional roof antenna equipped with a Low Noise Amplifier (LNA) with 30 dB gain and 1 dB Noise Figure (NF). The device used for sampling and replaying the signal was a HackRF One, an open source inexpensive SDR device [104]. The signal was sampled at 8 MHz with 8-bit I/Q quantization.

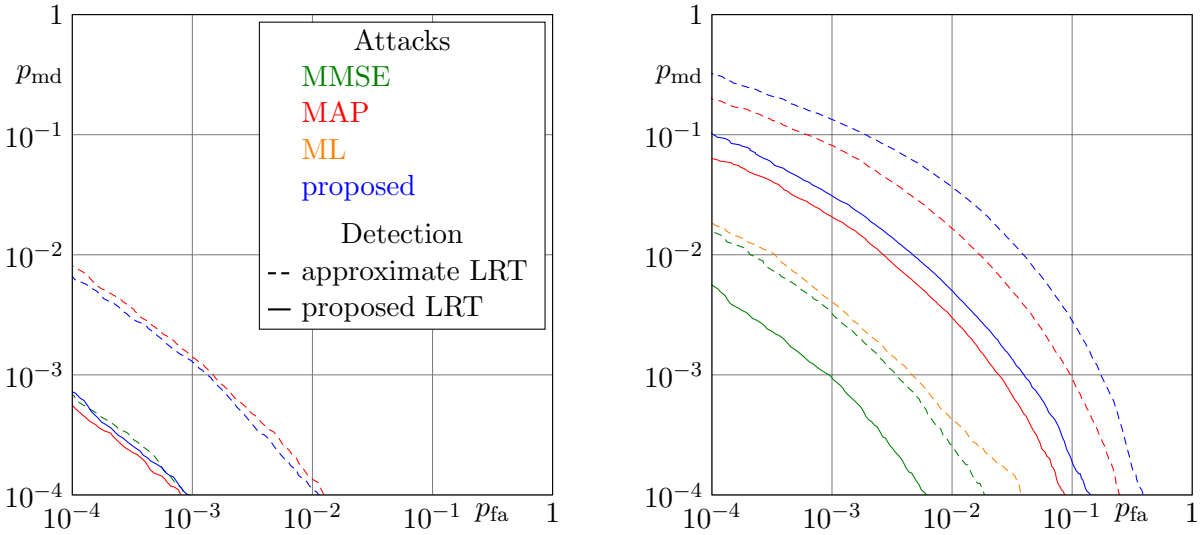


Figure 4.9: ROCs for the LRT detection method proposed in this paper and the one in [30], with the optimal SCER attack proposed here and those considered in [30]. System parameters are consistent with NMA on Galileo E1 OS: $(C/N_0)_{\text{rec}} = 48\,\text{dB}$, $T_{\text{d}} = 0\mu\text{s}$, $T_w = 4\,\text{ms}$, $T_{\text{f}} = 1.6\,\text{s}$. In (a) $(C/N_0)_{\text{att}} = 51\,\text{dB}$, while in (b) $(C/N_0)_{\text{att}} = 54\,\text{dB}$.

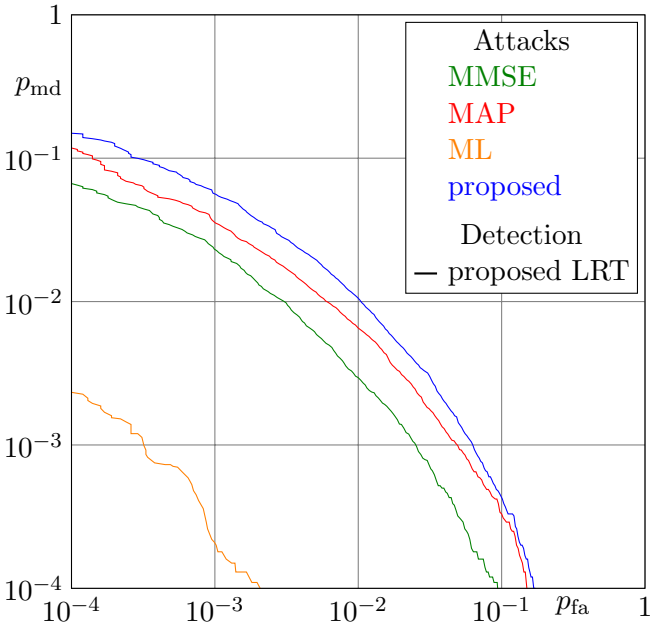


Figure 4.10: ROCs for the GLRT detection method proposed in this paper, with the optimal SCER attack proposed here and those considered in [30]. System parameters are consistent with SCE on GPS P(Y): $(C/N_0)_{\text{att}} = 51\,\text{dB}$, $(C/N_0)_{\text{rec}} = 48\,\text{dB}$, $T_w = 2\,\mu\text{s}$, $T_{\text{f}} = 0.8\,\text{ms}$, $T_{\text{d}} = 0\mu\,\text{s}$.

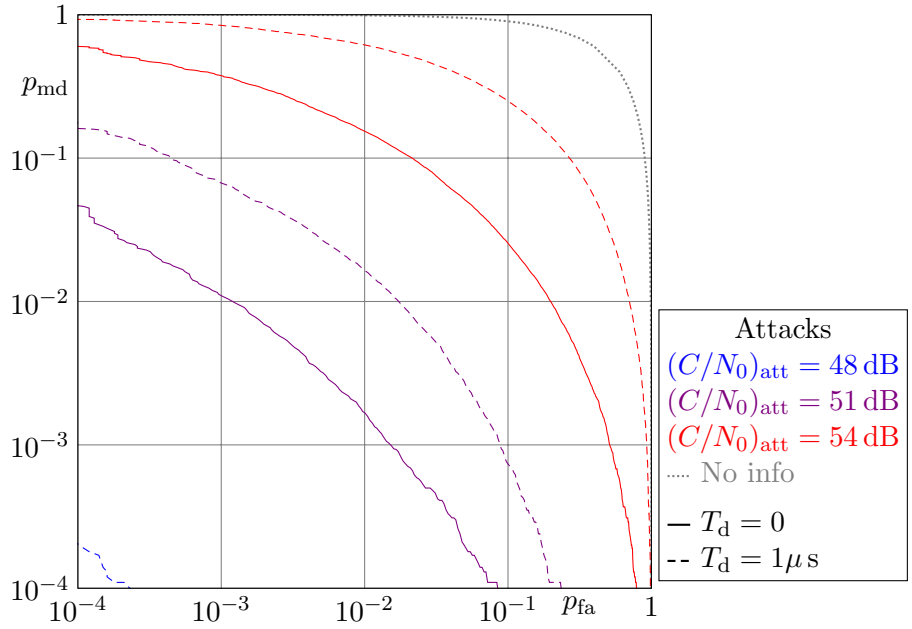


Figure 4.11: ROCs for the GLRT detection method proposed in this paper, with the optimal SCER attack proposed here and those considered in [30]. System parameters are consistent with NMA on Galileo E1 OS: $(C/N_0)_{\text{rec}} = 48\,\text{dB}$, $T_w = 4\,\text{ms}$, $T_{\text{f}} = 1.6\,\text{s}$.

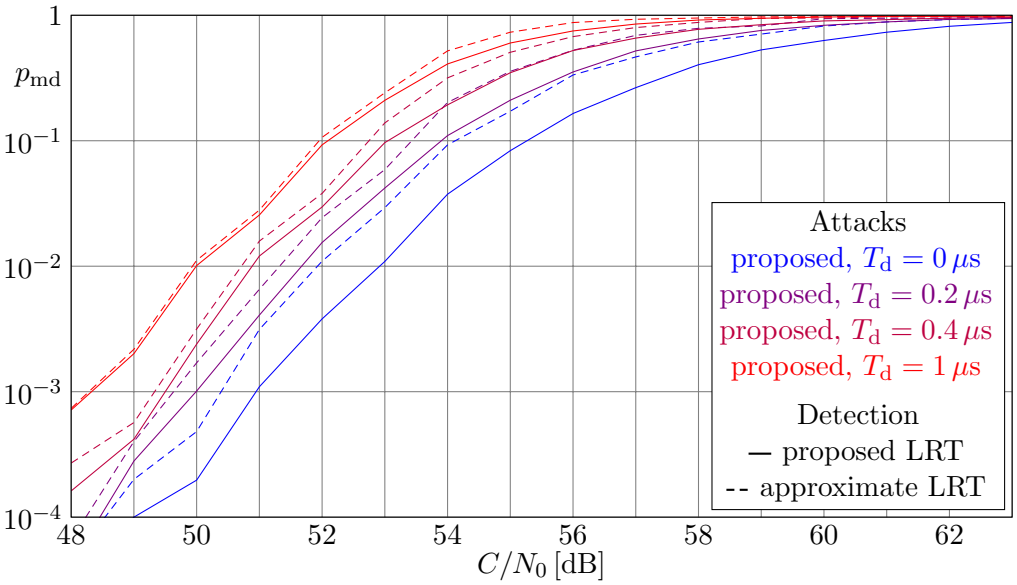


Figure 4.12: Probability of missed detection as function of $(C/N_0)_{\text{att}}$ for the LRT detection method proposed in this paper and the one in [30], for fixed $p_{\text{fa}} = 10^{-2}$, with the optimal SCER attack proposed here. System parameters are consistent with SCE on GPS P(Y): $(C/N_0)_{\text{rec}} = 48\,\text{dB}$, $T_w = 2\,\mu\text{s}$, $T_{\text{f}} = 0.8\,\text{ms}$.

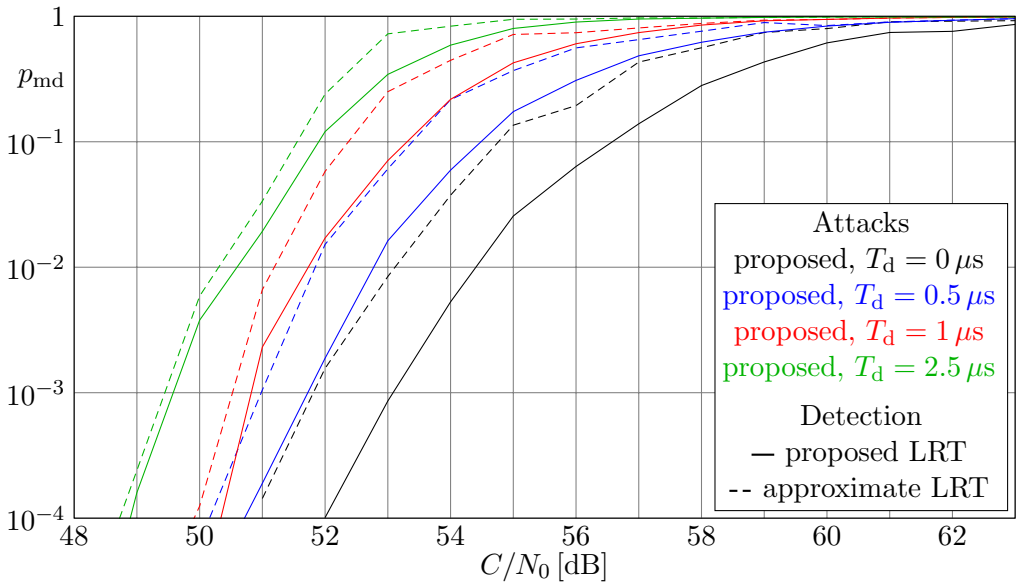


Figure 4.13: Probability of missed detection as function of $(C/N_0)_{\text{att}}$ for the LRT detection method proposed in this paper and the one in [30], for fixed $p_{\text{fa}} = 10^{-2}$, with the optimal SCER attack proposed here. System parameters are consistent with NMA on Galileo E1 OS: $(C/N_0)_{\text{rec}} = 48\,\text{dB}$, $T_w = 4\,\text{ms}$, $T_{\text{f}} = 1.6\,\text{s}$.

2. *Dynamic scenario:* in this case the signal is the clean dynamic acquisition of the Texas Spoofing Test Battery (TEXBAT) dataset [105]. This signal was acquired with a sampling rate of 25 MHz and 16 bit I/Q quantization, using a National Instruments PXIe-5663 Vector Signal Analyzer (VSA). The replay was performed using the HackRF One SDR.

Both the clean signal recorded from the antenna and the generated spoofed signals were replayed to a Septentrio PolaRx4 PRO with standard configuration.

For the sake of simplicity, when replaying the SCER signal, no authentic signal was present. The goal of the experiment was to verify whether the increase in the noise introduced by the attack was enough to prevent a professional receiver from decoding the navigation data and tracking the signal. The capture of the tracking loop was not implemented, since this would increase the complexity of the attack with respect to timing issues and power levels, but would not affect the attack strategy.

The employed estimator was the MAP (4.13). A zero delay attack was implemented, i.e., with $T_{\text{d}} = 0$ in (4.7), such that the attacker starts replaying the signal immediately after receiving the first sample of the authentic signal. Clearly, this initial estimation is rather poor and introduces a large amount of noise after the bit transition; however, this was used to evaluate the performance in the most challenging setting for the attacker. For the same reason, all 20 PRN repetitions of the C/A symbol were treated as independent in order to maximize the uncertainty for the attacker and the introduced noise. Due to the unreliable estimation of the first samples, a trivial extension of the attack is to start by using a few random samples, in order to generate a signal that arrives in phase at the correlation peak, producing a completely synchronized attack. In both experiments, the receiver was able to decode the navigation data and compute the PVT solution with the spoofed signal.

Two measures were defined to quantify the effectiveness of the attack:

- *Convergence time:* the time needed for the attacker estimation to stably reach the correct value, which can be written as:

$$C_t(n) = \max\{t \in [nT_w, (n+1)T_w],\ \text{s.t. } \hat{w}_n(t) \neq w_n\} - nT_w \tag{4.20}$$

In Fig. 4.14a the CDF of C_t is reported for different elevation angles in the two scenarios considered.

- *Correlation reduction:* due to the initial uncertainty in the attack estimate, the victim receiver will observe a normalized reduction in the correlation peak:

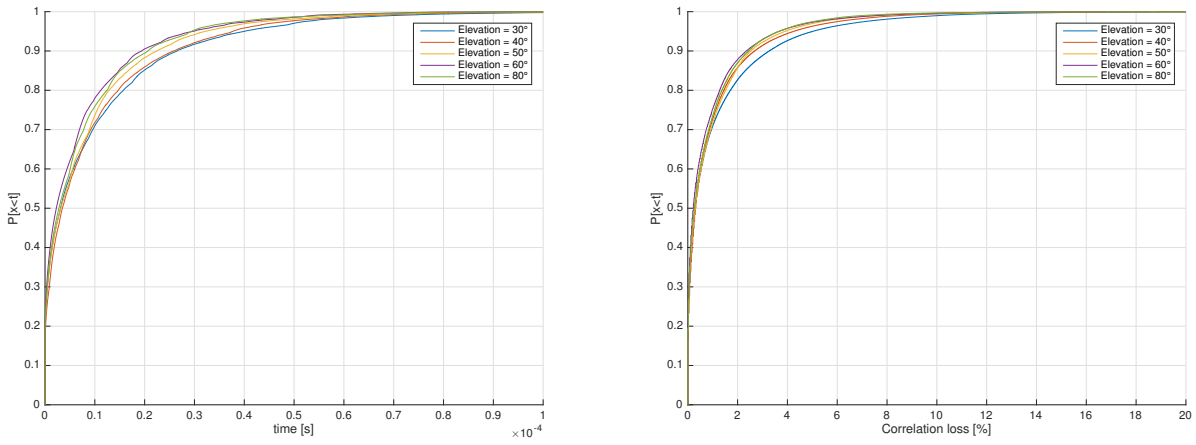
$$\Delta C(n) = \frac{\left|\int_{nT_w}^{(n+1)T_w} \tilde{y}(t)\tilde{s}(t)\,dt - \int_{nT_w}^{(n+1)T_w} y(t)\tilde{s}(t)\,dt\right|}{\left|\int_{nT_w}^{(n+1)T_w} y(t)\tilde{s}(t)\,dt\right|} \tag{4.21}$$

$$= \frac{\left|\int_{nT_w}^{(n+1)T_w} \hat{w}_n(t)\tilde{s}^2(t)\,dt - \int_{nT_w}^{(n+1)T_w} w_n(t)\tilde{s}^2(t)\,dt\right|}{\left|\int_{nT_w}^{(n+1)T_w} w_n(t)\tilde{s}^2(t)\,dt\right|} \tag{4.22}$$

$$= \frac{\left|\int_{nT_w}^{(n+1)T_w} (\hat{w}_n(t) - w_n(t))\,\tilde{s}^2(t)\,dt\right|}{\left|\int_{nT_w}^{(n+1)T_w} w_n(t)\tilde{s}^2(t)\,dt\right|} \tag{4.23}$$

In Fig. 4.14b the CDF of $\Delta C(n)$ is reported for different elevation angles in the two scenarios considered.

It can be seen from Fig. 4.14a that the time required to achieve the correct estimation with high probability (e.g., $> 95\%$) is in the order of 50 μs, which is only a small fraction of the symbol period for GPS C/A (20 ms) or Galileo E1B (4 ms). Therefore, the detection strategy should focus on this small fraction of each unpredictable symbol. Similar results were obtained in the dynamic scenario.



(a) (b)

Figure 4.14: CDF of (a) convergence time of SCER estimation and (b) the correlation reduction due to SCER attack in the static scenario, using MAP estimator.

4.4.6 Suboptimal detection scheme

During the implementation of the SCER attack discussed in the previous section, the difficulty of implementing the optimal LRT/GLRT was assessed. This difficulty comes from various sources, mainly, information of the received signal, which cannot be known unambiguously, is required. This information includes the noise variance of the signal. The receiver only has access to an estimation of

the carrier to noise ratio, which to a certain extent can be influenced by the attacker without being detected. In order to obtain sufficient information from the noisy signal, the receiver shall accumulate energy for a long period. This becomes increasingly difficult in an environment where measurements of the signal can change rapidly (e.g., due to multipath or receiver dynamics, especially for low elevation satellites). This makes it hard to use the correct average value and covariance matrix [30] in the computation and renders infeasible to precompute the covariance matrix for all the possible cases of interest. These issues significantly complicate the use of the optimal SCER detection strategy in the real world and motivates for a suboptimal detection strategy that is more robust and practical.

A simpler detection strategy is presented in [106], requiring less information on the received signal but achieving suboptimal performance. This proposal stores the first samples of each unpredictable symbol and evaluates the correlation with a local replica computed after the demodulation of the NMA This proposal demonstrates the feasibility of detecting SCER attacks in an AWGN channel; however, performance in realistic conditions were not evaluated.

An extension of this detection strategy includes dividing the symbols into bins and computing the correlation for each bin in addition to the evaluation of the accumulated correlation on the first part of the unpredictable symbols. If the signal is authentic, every bin will have the same average correlation value, while if the signal is generated by a SCER attack a smaller correlation for the initial bins is expected.

An example of the result of the suboptimal detection strategy, computed in the dynamic scenario discussed above and with the corresponding zero-delay SCER generated signal with MAP estimator, is reported in Fig. 4.15. The left column represents the case in which a bin width of 10 μs is used, while in the right column the bin width is set to 100 μs. Fig. 4.15a-Fig. 4.15b show that for authentic signals all the bins have similar correlations. In Fig. 4.15c-Fig. 4.15d the spoofed signal was used and the tracked SV signal had a high C/N_0. Fig. 4.15e-Fig. 4.15f illustrate the case when the tracked SV signal has a low C/N_0. As expected, the shorter the bin, the more evident the loss of correlation on the initial bins; however, the estimate also becomes noisier. For this reason a trade-off shall be found.

If instead of a zero-delay attack the attacker is able to gain even a little time advantage by exploiting the receiver clock uncertainty, then the situation dramatically worsens for the receiver, which quickly loses the ability to detect the initial attacker uncertainty at all.

It is also clear from Fig. 4.15 that it is hard to define an optimal spoofing detection threshold for the receiver. Indeed, the correlation level heavily depends on the C/N_0 and thus on the noise variance of the received signal. As already discussed, the receiver cannot reliably determine his actual C/N_0 nor check if the estimated value corresponds to the expected one, predominantly due to effects linked to the environment. While it might be possible for a static receiver to do this in open sky conditions (e.g., where historical values or a model of the average C/N_0 based on the SV elevation could be used), it is not considered feasible for dynamic receivers. Furthermore, an attacker could easily influence the C/N_0 estimation by artificially introducing noise in the generated signal or intentionally flip (invert the phase) the generated signal for a fraction of the bin duration, to lower the measured correlation at the receiver side. In this way, the attacker could reduce the distance between the first bins and the rest of the symbol, relaxing the need for a perfect estimation since the beginning.

It is possible to formulate this attack as follows: the victim receiver makes use of the correlation based detection strategy, using N bins. Let C_n be the correlation value in the n-th bin and $\boldsymbol{C} = [C_1, \ldots, C_N]$. The detection strategy is to accept the signal as authentic if $\boldsymbol{C}$ lies within some predetermined set $\mathcal{C}_0$. The attacker, that is supposed to know N, aims at inducing a flat correlation observed by the receiver for each bin. In order to do this he can perform a training phase in which he obtains an estimation of the typical correlation shape by performing a dry run of the SCER attack and of the detection strategy. In this way he can achieve an estimation similar to the one in Fig. 4.15c. Now the attacker can balance the effect of the attack. Let us define N_s as the number of samples contained in each bin, A_s as the sample amplitude corresponding to the PRN processed, C_1 as the

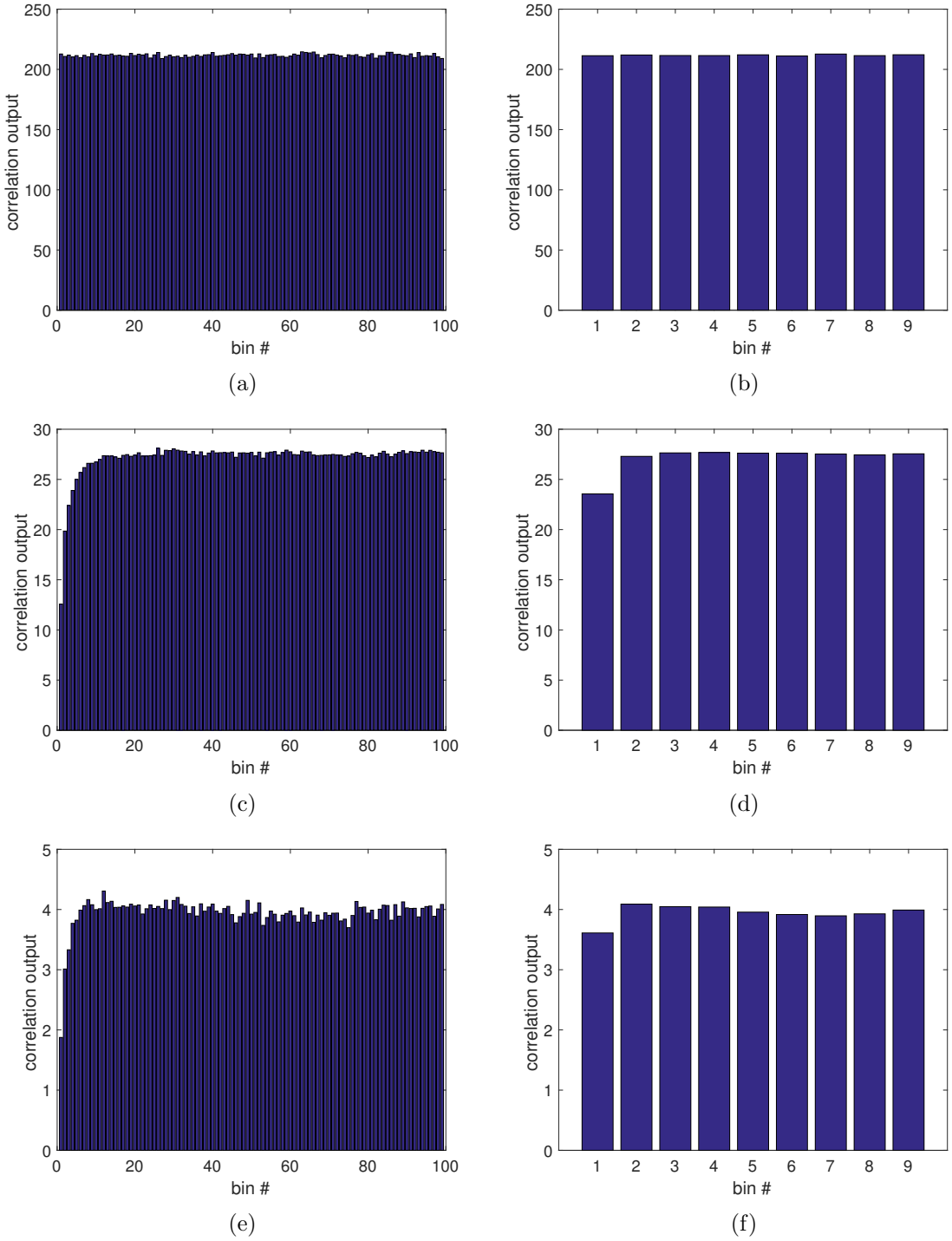


Figure 4.15: Suboptimal SCER detection output for the dynamic scenario. The bins width is 10 and 100 μs respectively in the left and right column. The signal used was: authentic signals with high C/N_0 in (a) - (b); spoofed signal with high C/N_0 in (c) - (d); spoofed signal with low C/N_0 in (e) - (f).

correlation value of the first bin and C_n as the correlation obtained in the n-th bin. The goal of the attacker is to reduce the average correlation obtained in the bin $n \geq 2$ to C_1. The number of samples

that must be flipped, N_{flip_n}, can be computed as:

$$N_{flip_n} = \frac{C_n - C_1}{2C_1} \tag{4.24}$$

The generated signal presents a flat correlation over all the bins when transmitted; the knowledge of the receiver noise statistic is not needed, because this will be equally distributed over all the bins.

Table 4.1 details synthetic results and the corresponding parameters of the attack used to obtain the flat correlation shown in Fig. 4.16 in the previously described dynamic scenario. It is possible to see that artificially flipping only few μs of signal for each bin it is possible to obtain a flat correlation, aligned with the first bin. The small number of flipped chips makes detection very challenging, as the attack cannot be easily differentiated from effects linked to the environment (e.g., multipath).

C/N_0	T_{bin} [μs]	n_s	A_s	C_1	C_n	N_{flip_n}	T_{flip} [μs]	%
high	100	2500	27.6	58750	69000	185	7.4	7.4%
high	10	250	27.6	3145	6900	68	2.7	27%
low	100	2500	4	8750	10000	157	6.24	6.2%
low	10	250	4	58750	69000	68	2.7	27%

Table 4.1: Summary of the parameters used to balance the effect of the SCER attack.

C/N_0	T_{bin} [μs]	N_{flip_1}	N_{flip_2}	N_{flip_3}	N_{flip_4}	N_{flip_5}	N_{flip_6}	N_{flip_7}	N_{flip_8}	N_{flip_9}	$N_{flip_{10}}$	$N_{flip_{11}}$	$\cdots$
low	10	0	48	55	63	64	67	67	67	67	67	67	$\cdots$
high	10	0	46	55	59	62	64	65	66	66	66	67	$\cdots$
low	100	0	157	157	157	157	157	157	157	157	-	-	-
high	100	0	185	185	185	185	185	185	185	185	-	-	-

Table 4.2: Number of samples flipped in the n-th bin in order to balance the SCER attack. The result is shown in Fig. 4.16.

The drawback of this attack strategy is the degradation of the C/N_0. In order to reduce this effect, the attacker can attempt to maximize the average correlation of the first bin C_1, for example, using a time-varying strategy that reduces the power level of the first samples where he has the maximum uncertainty, and transmitting the last samples of the first bin at a higher power. A pictorial example is reported in Fig. 4.17, where the power level in the first bin starts at zero and increases over the time. After the first bin, in the bin 2, 3 and 4 the power level is set to a stable value that should match the expected value of the correlation of the first bin. Clearly the attacker has some constraint on the power level used, both for hardware limitations and to avoid being easily detected. Indeed, several anti-spoofing techniques make use of the received power, for instance monitoring the AGC level [14].

The long symbol periods of open service GNSS signals, yield a significant opportunity for the attacker. After reaching a reliable estimate $\hat{w}_n(t)$, the attacker can generate a signal that almost perfectly resembles the legitimate signal. The attacker is able to exploit the long integration time at the receiver side to hide the initial uncertainty. Moreover, the noise level in the signal can be artificially increased in order to hide the initial uncertainty and make it comparable to the noise of the signal. Due to the long symbol period, the receiver will still track and demodulate the symbol correctly.

It is worth noting that with more recent signals such as the Galileo E1B or GPS L1C, which make use of channel coding to reduce the BER, the attacker may even have his incorrect estimations

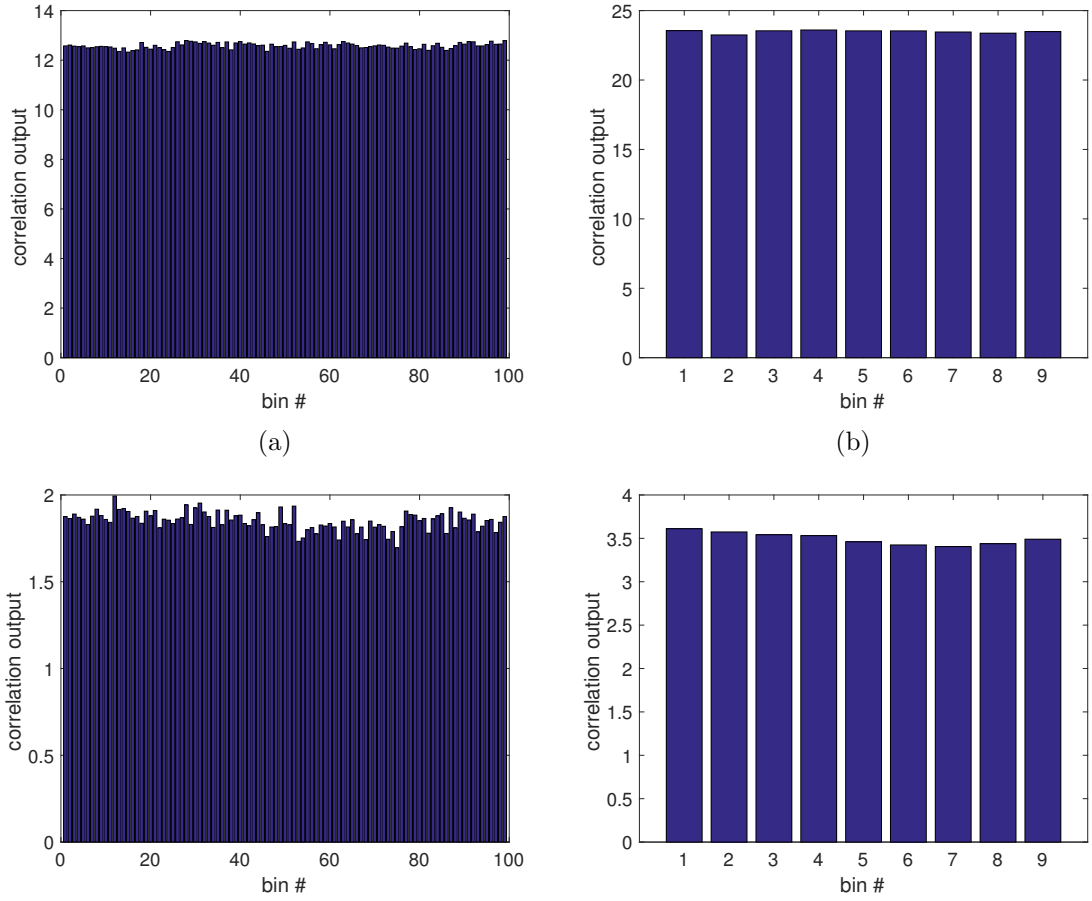


Figure 4.16: Suboptimal SCER detection output for the dynamic scenario with balance attack.

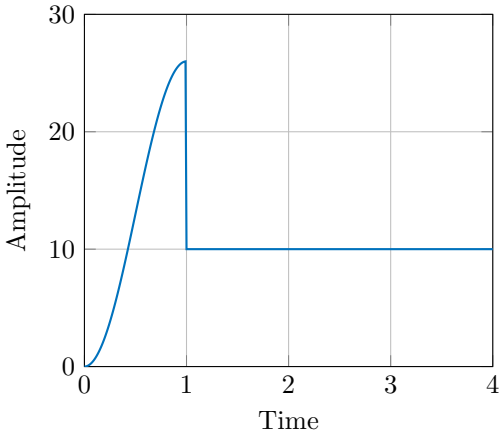


Figure 4.17: Example of time-varying power level for SCER attack against correlation based detection strategy.

corrected by the victim decoder itself and the FEC redundancy can be leveraged to mount a FEA attack [74].

A method to limit the effectiveness of the attack is to increase the difficulty of estimating unpredictable symbols. This can be done by reducing the per symbol energy, either by reducing the

transmission power or the symbol length.

Use of GNSS signals with much lower energy per symbol (higher chipping rate) can significantly increase the difficulty of an attack. For example the GPS P(Y) signal, with a chipping rate 10 times higher than the C/A code. The fact that the P(Y) spreading code is not public (i.e. modulo-2 sum of P-code and encrypting code), thus an attacker cannot exploit the processing gain, and the observed energy per symbol is significantly reduced. This, coupled with higher chipping rate, significantly increases the difficulty of the attack.

In the next section, a technique is presented that exploits the characteristics of the GPS P(Y) signal to restrict the opportunities for the attacker.

Future investigation on this topic may concern the use of *quickest detection* theory [107] for SCER detection. Quickest detection is a statistical framework that aim at minimize the time between the change in the statistical distribution of a signal or of a time series under analysis. The cumulative sum (CUSUM) is one of the most popular test in this class. Its use was already investigated in the GNSS context for both interference detection [108], and spoofing detection [109] using multiple antennas.

4.5 Semi-codeless techniques for anti-spoofing

The GPS C/A code is the most widely used signal by civilian GNSS receivers today. Historically, this signal was used to facilitate the handover process from C/A to P(Y) code tracking, allowing the receiver to determine the correct P-code setup parameters and whether spreading code encryption was active. Today direct acquisition is possible for military users; however, the handover process is still used by some semi-codeless receivers in order to obtain multi-frequency measurements for high-precision applications. The P(Y) code is a modulo-2 sum of the P-code and W-code, an encrypting code which is not known to unauthorized users. The P code chipping rate is 10.23 Mchip/s and the W chipping rate is 20 times lower, 511.5 Kchip/s.

The GPS anti-spoofing mechanism not only limited the access to the higher-precision precision signal, but also to the L2 frequency where only the P(Y) signal was transmitted.

In order to have access to a second frequency, providing the possibility to correct errors induced by the ionosphere, semi-codeless techniques allowed carrier phase measurements to be made on the L2 frequency without knowledge of the secret W code [110]. Simpler techniques (codeless), involved squaring of the received signal; however, the squaring operation also increases the noise (reducing the signal to noise ratio). More sophisticated techniques exploited knowledge of the public P-code before squaring, maintaining uncertainty on the W-code. By wiping off the P-code, a 20 times reduction of the signal bandwidth was obtained. Thus by filtering the signal with a bandpass filter before the squaring operation, the noise could be reduced by 13 dB. These techniques are commonly referred to as P-code aided squaring.

Several additional techniques were proposed in the literature for combining the P(Y) signal transmitted in both the L1 and L2 frequency, allowing a further performance improvement. An interesting feature of these techniques is that instead of simply squaring the W code, they perform an estimation of the W-code, using a range of different estimator techniques.

Our work focuses on the first two techniques discussed as they only require the L1 signal component. We propose to exploit semi-codeless techniques for the purpose of anti-spoofing. Extension to multi-frequency receivers is trivial and may allow to achieve better performance.

The following assumptions are made for the anti-spoofing technique presented in this Section:

- The receiver is static, in an open-sky environment tracking high-elevation satellites. It is assumed that the C/N_0 is relatively stable and that the variations due to effects of the local environment such as multipath are limited.

- The receiver may be under attack by a spoofer. The task of the receiver is to determine whether it is under attack and to provide the capability to distinguish between authentic and spoofed signals.
- It is assumed that the attacker is not able to block legitimate signals (e.g., by unplugging the victim receiver antenna and connecting it directly to the spoofer). As a consequence, the technique may not be suitable for applications where the user himself is the attacker (i.e., self-spoofing).
- It is assumed the signal is unpredictable, such that an attacker is not able to generate a valid signal before the transmission of the authentic one from the satellite.

A high-level overview of attack strategies is described below:

- *Simple signal generation:* a non-sophisticated attacker may generate the C/A signal component only. The use of semi-codeless techniques allows the receiver to detect the absence of the P(Y) component or whether the fixed power ratio between the component is verified.
- *Complex signal generation:* a sophisticated attacker may generate both C/A and P(Y) components as per the Interface Control Document (ICD). Because the W code is not public, the victim is unable to directly check if the received signal is modulated by the correct W code. There are some techniques that attempt a cross-check between receivers [94] or send the sampled RF signal to a secure server, which has access to the military code to perform the PVT computation. If the generated signals can reach the receiver antenna synchronized with the authentic signals, the detection would be based on the difference of the W-code. In [94] the idea of extracting the W-code through a semi-codeless receiver and the comparison with an estimation coming from a ground infrastructure equipped with a high gain antenna is presented.
- *Meaconing:* the simplest way to spoof the signal with the authentic W code is to perform a meaconing attack. The attacker receives the signal, waits for a desired delay, and then retransmits the signal towards the victim receiver. The signals will not be aligned at the receiver antenna, thus at least two correlation peaks (both on C/A and P(Y)) would be found in the acquisition, both with the correct W-code. In the case of meaconing, ranges can only be delayed (not anticipated). Furthermore, this type of attack would result in a spoofed signal that is noisier than the authentic one.
- *SCER:* a sophisticated attacker may try to reduce the noise on the spoofed signal by performing an estimation of the W-code as described in Section 4.4. Even with this type of attack, a delay in the generation of the spoofing signal is introduced. Due to the very short bit duration (about $2\mu s$), the attacker is constrained in the amount of energy that can be accumulated in this duration without significant antenna gain (e.g., steerable dish antennas), see Fig. 4.14a. The difference with the SCER attack on the C/A is the symbol duration and therefore the amount of energy that can be accumulated. This permits a synchronized attack by generating random samples before sufficient energy has been accumulated and hiding this by adding noise once a reliable estimate has been obtained so that this type of attack cannot be easily differentiated from noise induced from the environment. Due to the limited energy that can be accumulated for a short P(Y) chip duration, techniques for hiding the attack strategy are very limited. The short bit duration therefore significantly reduces the effectiveness of a SCER attack, where the need for a higher antenna gain would make an attack more expensive and possibly visible (less covert).

Considering a signal with an unknown code and very short bit duration (e.g. W-code bit), the degrees of freedom for the attacker to be able to generate an undetectable aligned spoofing signal with the correct code are very limited.

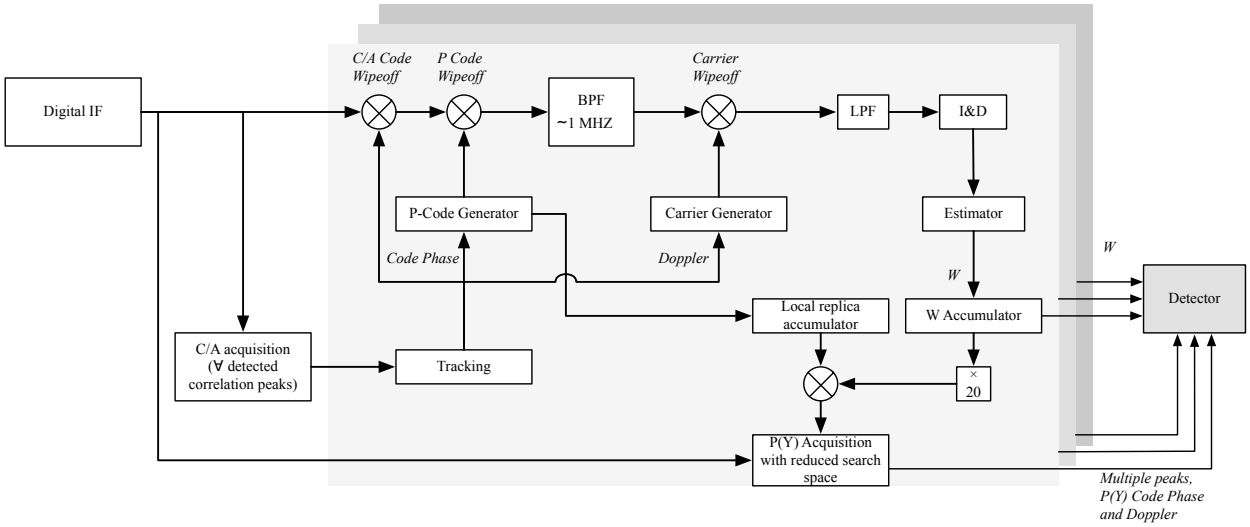


Figure 4.18: Proposed autonomous anti-spoofing scheme.

4.5.1 P(Y) acquisition with reduced search space

The first technique we propose consists in exploiting the semi-codeless tracking methods to obtain an estimate of the received W code, and to verify the consistency of the P(Y) code phase and carrier frequency with the C/A one. The concept is shown in Fig. 4.18. The digitized signal is acquired and tracked using the C/A component. The output of the tracking loop is used to wipe-off both ranging codes (C/A and P) and the result is filtered to reduce noise. Then, the carrier phase is removed using the PLL output. The only component left in the signal is the W code. Integrating at the W code rate and using an estimator (e.g., the MAP estimator discussed in the SCER section) allows to recover the secret W bits. Due to the independence of the cryptographical bits, no processing gain can be leveraged and thus there is no advantage in using longer integration time, thus each bit can be estimated independently. The probability of correct bit estimation will degrade sharply below a certain C/N_0. If the estimation is performed independently for each W chip, selecting the combination with the maximum energy detected among all the combinations used, this becomes a semi-coherent integration that provides advantage over a non-coherent integration [18]. Clearly, the reliability of the estimation depends on the C/N_0, the antenna gain, the estimator used and the noise figure of the receiver.

4.5.1.1 W code coherence

When a W bit sequence estimation is available, it is passed to a detector that compares the estimation coming from different tracking loops locked on different correlation peaks in order to verify if all the tracked replicas are modulated by the same secret spreading code for that PRN. In order to compare between two estimations the cross-correlation is computed. We can distinguish three cases:

(a) both the signals are authentic, e.g., LOS and multipath of the authentic signal: in this case the cross-correlation should show a correlation peak.

(b) both the signals are spoofed, e.g., LOS and multipath of the spoofed signal: even in this case the cross-correlation should show a correlation peak, rendering the detection ineffective. For this reason we assume that at least one of the peak corresponds to an authentic signal.

(c) one signal is authentic and the other is spoofed.

In order to discriminate between case (a) and (c), a characterization of the cross-correlation result is needed. Due to the independence of the W bits, a multibit statistic is just an extension of the single bit one. Let us denote by p the equivalent BER of the receiver, and by q the equivalent BER of the attacker. These include the channel effects, the antenna gain and the estimation strategy.

Let x be the random variable that represents the product of two W bit estimations. For the sake of simplicity we can assume a hard detection estimation that represents a pessimistic assumption for the performance. The pmd of x when both the signals are legitimate $p_{x|ll}(a)$ and when a signal is legitimate and the other one is spoofed $p_{x|ls}(a)$ can be written as:

$$p_{x|ll}(a) = \begin{cases} p^2 + (1-p)^2 & a = 1 \\ 2p(1-p) & a = -1 \end{cases} \tag{4.25}$$

$$p_{x|ls}(a) = \begin{cases} p^2 q + q(p-1)^2 + 2p(p-1)(q-1) & a = 1 \\ (p-1)^2(1-q) + p^2(1-q) + 2pq(1-p) & a = -1 \end{cases} \tag{4.26}$$

The detection strategy for determining between the two cases is a hypothesis testing problem based on the cross-correlation of the two estimations:

$$y = \sum_{n=1}^{N} x_n \tag{4.27}$$

This sum is a random variable itself and it is possible to write it as the sum of two Gaussian variables:

$$\begin{aligned} N_+ &= \sum_{n=1}^{N} \chi\{x_n = 1\} \\ &\sim \mathcal{N}\left(N p_{x|z}(1), N p_{x|z}(1) p_{x|z}(-1)\right) \end{aligned} \tag{4.28}$$

$$\begin{aligned} N_- &= \sum_{n=1}^{N} \chi\{x_n = -1\} \\ &\sim \mathcal{N}\left(N p_{x|z}(-1), N p_{x|z}(-1) p_{x|z}(1)\right) \end{aligned} \tag{4.29}$$

$$p_{y|z}(a) \simeq N_+ - N_- = N - 2N_- \tag{4.30}$$

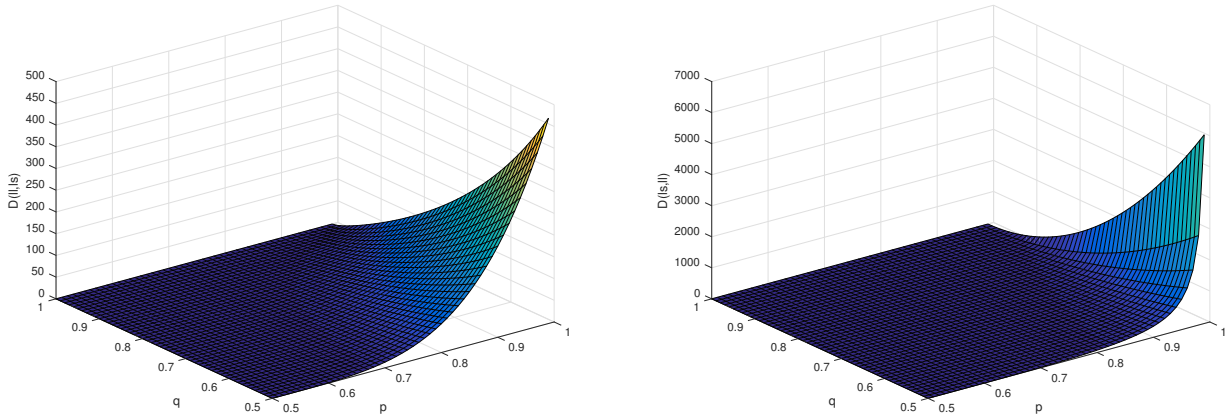
where z can be ll or ls. The K-L divergence between two Gaussian distributions $\mathcal{N}(\nu_a, \sigma_a^2)$ and $\mathcal{N}(\nu_b, \sigma_b^2)$ is given by

$$D(a,b) = \log\left(\frac{\sigma_a}{\sigma_b}\right) + \left(\frac{\sigma_a^2 + (\mu_a - \mu_b)^2}{2\sigma_b^2}\right) - \frac{1}{2} \tag{4.31}$$

When deriving $D(p_{y|ll}, p_{y|ls})$ and $D(p_{y|ls}, p_{y|ll})$ (see Fig. 4.19), is seen that as $p \to 0.5$ both divergences are close to 0 irrespective of q. For $p \to 1$, the distinguishability depends on q: for q close to 0.5 it is very easy to detect the attack, while for q close to 1 it is more difficult.

It is possible to select the target p, q and compute the outer bound in the detection probability using the LRT, that is the strategy that yields the minimum p_{md} for any given constraint on p_{fa}, and *vice versa* [102, §3.3][98] if both the statistics of the legitimate and spoofed signal are known (i.e., if the victim is aware of the particular strategy adopted by the spoofer). If the spoofer strategy is not known, a GLRT strategy shall be used but will lead to worse results.

Based on the BER of both the attacker and the receiver, it is possible to find the minimum observation time to achieve the desired performance in terms of false alarm probability p_{fa} and of missed detection probability p_{md}. The bigger the attacker advantage over the victim receiver, the longer the necessary minimum observation time, which in turn leads to a longer TBA.



(a) (b)

Figure 4.19: K-L divergence for 1 ms of W bit sequence observation.

An example is reported in Fig. 4.20, for (a) $p = 0.75$, $q = 0.9$ and (b) $p = 0.55$, $q = 0.9$. It is evident that when the attacker advantage is limited, the detection mechanism requires a short observation time; while as his advantage increases the receiver needs to accumulate more W chips, requiring a much longer time before being able to distinguish between the authentic and spoofed signals. An advantage of working with the W secret code is that due to its high rate of 511,5 Kbit/s, the 6 Mbit taken into account correspond to an observation period slightly shorter than 12 seconds.

In this section a detection statistic between two peaks was presented. The extension to a detection statistic that takes into account distorted (or multiple) auto-correlation peaks, possibly achieving better results by leveraging more signal features (e.g., correspondence to a multipath model), is left for future work.

4.5.1.2 P(Y) Acquisition

When a sufficiently long W sequence is estimated (in the order of tens of milliseconds), a second check can be performed. This check is a direct P(Y) acquisition. At this stage, the receiver has achieved a good time synchronization and knows which W sequence shall be used to build the local replica, thus there is no need to perform the acquisition over a big search space, and the computational complexity can be reduced. On the other hand, a larger search space, improves the anti-spoofing performance. A tradeoff between the search space dimension and the search resolution shall be found. Moreover, the time required to perform this acquisition is not critical and can take up to some seconds, alleviating the computational requirement. Clearly, the performance depends on the BER on the W code estimation, and the worse the estimation performance, the longer the required integration time to achieve the desired detection capability.

The output of the acquisition stage are three measures: estimated carrier frequency, estimated code phase, and the number of correlation peaks. Two types of checks can be performed on these outputs. The first check is on the number of correlation peaks and on their relative position. If the signal is LOS, only one peak should be found. If LOS and multipath are present, a distorted correlation peak or secondary smaller delayed peaks can be found. The presence of two strong peaks very distant from each other, may indicate a meaconing attack. The estimated code phase and carrier frequency can be compared with the ones obtained from the C/A acquisition in order to check the coherence of the measures. A mismatch may also indicate a spoofing attack. The value of the correlation peak can also be compared to that of the C/A in order to check the coherence with the fixed power relationship.

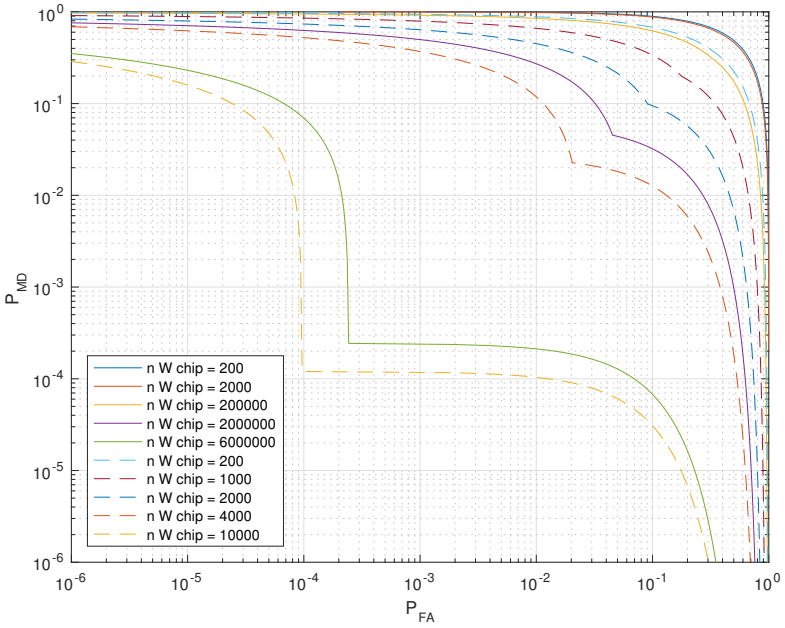


Figure 4.20: Outer bound of achievable performance for different p, q and observation time. $q = 0.9$, while $p = 0.55$ for solid lines and $p = 0.75$ for dashed lines.

4.5.2 Estimation of residual energy

A second way to exploit semi-codeless tracking techniques is an iterative estimation and removal process.

The carrier-phase measurements are typically used in surveying applications due to the improved accuracy that they achieve with respect to code-phase measurements. Indeed, code-phase measurements are more affected by multipath and are in general much noisier.

For these reasons, a second integrity check can be performed comparing carrier-phase measurements obtained from a different iteration of the processing.

The scheme is shown in Fig. 4.21. The initial part of the processing is equivalent to the one proposed in Fig. 4.18. After the carrier wipe-off, besides the W code, other terms such as multipath or a spoofed signal are also present in the signal. Recognizing this, the processing is repeated on this residual energy. If the receiver is tracking a spoofed C/A signal, the residual energy will contain the non spoofed C/A and P(Y), which will be coherent between them. As an example, we can think of a P code aided squaring (also known as semi-codeless squaring) and to use long coherent integration time to obtain a low noise estimation of the Doppler frequency.

In the ideal case of LOS and of perfect tracking by the first tracking loop, no signal is present after the wipe-off, so the second tracking loop should not be able to acquire and track the signal. If the receiver is under spoofing attack, the second tracking loop should acquire and track a signal. The estimated Doppler frequency shall be different form the first one, as indeed the spoofed is trying to influence the PVT computation of the receiver by changing the ranging. If there is multipath, the second tracking loop should be able to acquire and track a signal.

This approach suffers from an important drawback: the estimation (even more in the case of squaring) introduces some interference into the remaining part of the signal. These terms will also include cross-correlation terms, and this may degrade the sensitivity of the scheme. This issue is even more evident if a codeless technique is applied, because the goal of the detection is to distinguish the Doppler frequency of all the legitimate SVs in view from possible unknown spoofing terms.

On the other hand, even if suffering from squaring losses, the receiver complexity is reduced with

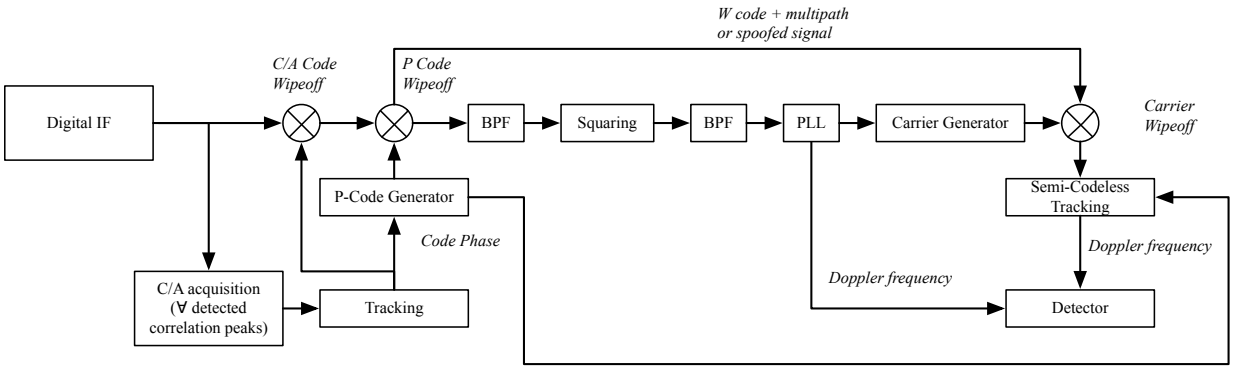


Figure 4.21: Proposed autonomous anti-spoofing scheme.

respect to relative to the semi-coherent estimation of the W code presented in the Section 4.4. For these reasons the determination of an optimal detection strategy is postponed to future work.

4.6 A signal level scheme based on integrity codes

Integrity Codes (I-code) are a mechanism to protect the integrity of broadcast messages without requiring a prior shared secret, that was proposed in [111]. The mechanism operates at the physical layer allowing detecting inconsistencies in the message received due to transmission of malicious message from an attacker. Let the transmitted signal $x(t)$ carry the legitimate navigation message m. The attacker sends out a signal $z(t)$ such that the received signal $y(t) = x(t) + z(t)$ will be decoded to a false message $m' \neq m$. A first use in GNSS proposed in [112] leverages

- *on/off* binary modulation with '1' being encoded to some waveform $x_1(t)$
- *unidirectional* error detecting codes (e.g., Manchester) so that any '0' $\rightarrow$ '1' modification can be detected
- a *randomized* waveform $x_1(t)$ to avoid '1' $\rightarrow$ '0' modification through cancellation with $z(t) = -x_1(t)$
- a simple *energy detection* receiver, no need to share randomness

4.6.1 Unidirectional error detecting code

The unidirectional error detecting code proposed in [111] is the Manchester coding, that encodes a message of length k bit to a sequence of length $2k$. In [113] a more efficient unidirectional coding is proposed, that outputs sequences of $k + O(\log_2 k)$ bit. The coded message consists of the original message, denoted by S_0, with the appendix of a series of segments $S_1, S_2, \ldots, S_\ell$. Each segment S_i have lengths k_i defined as:

$$k_i = \begin{cases} k & i = 0 \\ \lfloor \log k_{i-1} \rfloor + 1 & 1 \leq i \leq \ell - 2 \\ 2 & \ell - 1 \leq i \leq \ell \end{cases} \tag{4.32}$$

The segment S_i is the binary representation of the number of '1' bits in the preceding segment S_{i-1}. The receiver can check the integrity of the message by comparing the number of '1' in each segment with the number reported in the next segment. Since the adversary is only able to change '0' to '1', any modification will result inconsistent with the coding.

4.6.2 Randomization of the waveform

The use of I-code in GNSS is not straightforward due to the randomization process of the waveform. Indeed, this shall not destroy the orthogonality between PRNs for different SVs and affect the receiver design increasing the complexity.

The original concept [111] achieves randomization of the waveform by randomly changing the carrier phase multiple times in each symbol period. On one side, this makes it difficult for an attacker to guess the actual phase shifts and to generate a signal that arrives in anti-phase at the receiver antenna, but it is difficult to be tracked by a traditional PLL architecture. To simplify the receiver design, it is proposed to randomize the waveform by flipping some chips in the PRN code. More precisely, each chip is independently flipped with probability $p < 1/2$.

The signal generation is depicted in Fig. 4.22. The message (Fig. 4.22a) is first encoded using the unidirectional code (Fig. 4.22b), then each '1' bit is multiplied by the spreading code (Fig. 4.22c) while the '0' bit is left as a radio silence. In order to achieve randomization, some chips of the PRN are randomly flipped (Fig. 4.22d). Finally, the generated message is transmitted between two consecutive I-delimiters (Fig. 4.22e), bit sequences that violate the coding rule, in order to allow message synchronization. For the Manchester coding a simple I-delimiter is represented by the sequence '111000' that violates the maximum consecutive number of '0's and '1's that is two.

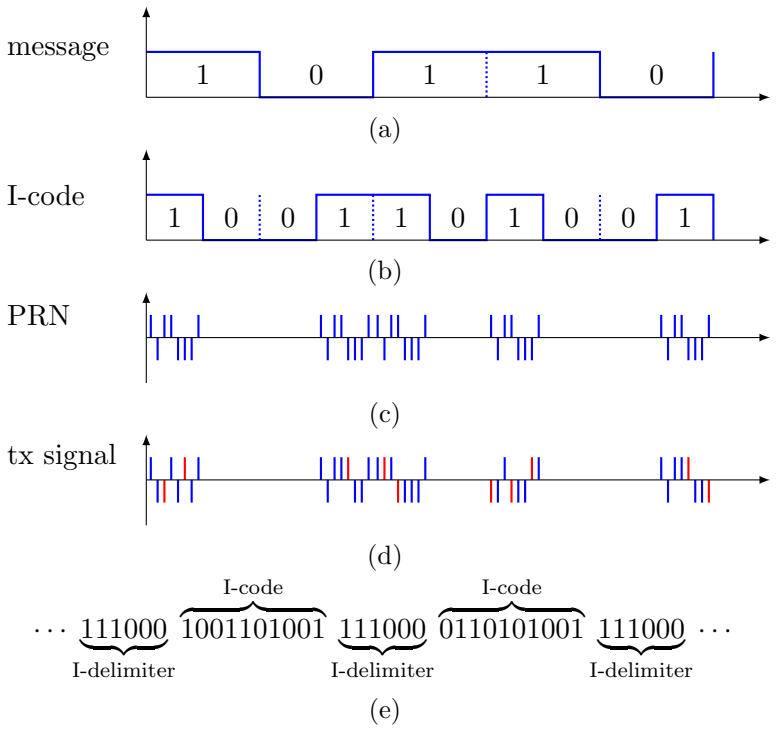


Figure 4.22: I-codes generation and transmission.

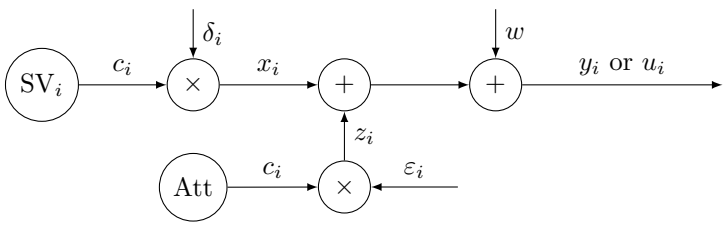


Figure 4.23: Randomization of the I-codes waveform through chip flipping.

Let us define $c_i(n)$ as the n-th chip of the spreading code associated to the SV_i. Every chip $c_i(n)$ of the spreading code $\boldsymbol{c}_i$ is flipped in accordance to a random variable $\delta_i(n) \in \{\pm 1\}$ with probability

$$P[\delta_i(n) = -1] = p \tag{4.33}$$

The SV sends the sequence $x_i(n) = \delta_i(n)c_i(n)$. The spoofer decide whether to flip or not the chips of $c_i(n)$ according to a random variable $\varepsilon_i(n) \in \{\pm 1\}$ with probability

$$P[\varepsilon_i(n) = -1] = q \tag{4.34}$$

and broadcast the resulting sequence $z_i(n) = \varepsilon_i(n)c_i(n)$. The received signal is

$$\begin{cases} \boldsymbol{y}_i = \boldsymbol{x}_i + \boldsymbol{w} & , \text{ for the legitimate signal} \\ \boldsymbol{u}_i = \boldsymbol{x}_i - \boldsymbol{z}_i + \boldsymbol{w} & , \text{ for the cancelled signal} \end{cases} \tag{4.35}$$

where $\boldsymbol{w}$ is the receiver noise. It is possible to rewrite $u_i(n)$ as

$$u_i(n) = \delta_i(n)c_i(n) - \varepsilon(n)c_i(n) + w_i(n) = (\delta_i(n) - \varepsilon_i(n))c_i(n) + w_i(n) = \alpha_i(n)c_i(n) + w_i(n) \tag{4.36}$$

where $\alpha_i(n)$ becomes a random variable that takes values in $\{-2, 0, 2\}$ with pmd:

$$p_{\alpha_i(n)}(a) = \begin{cases} p(1-q) & a = 2 \\ 1 - (p+q) + 2pq & a = 0 \\ q(1-p) & a = -2 \end{cases} \tag{4.37}$$

Intuitively, a lower probability of flipping the chip p leads to a better distinguishability among different PRNs but eases the prediction and the cancellation of the legitimate navigation signal by an attacker. Thus, on the one hand $p \to 0$ is the optimal case for the usability, while on the other hand $p \to 0.5$ is the optimal case for the security. In the following Section a security analysis will be presented, that can be used to design the system according to the desired performance.

4.6.3 Attack model and success probability

The attacker attempts to cancel $x_i(t)$ (randomized) by transmitting a randomized waveform $z_i(t)$ opposite to the PRN code where each chip is independently flipped with probability $q < 1/2$. The K-L divergence between the legitimate and the canceled signal $\mathbb{D}(\boldsymbol{y}_i||\boldsymbol{u}_i)$ due to the independent flipping process can be written as $L\mathbb{D}(y_i||u_i)$ where L is the number of chip considered. The pdf of the received signal canceled by the attacker can be written as:

$$f_u(a) = \sum_{m \in \mathcal{A}_\alpha} p_\alpha(m) f_w(a - m) \tag{4.38}$$

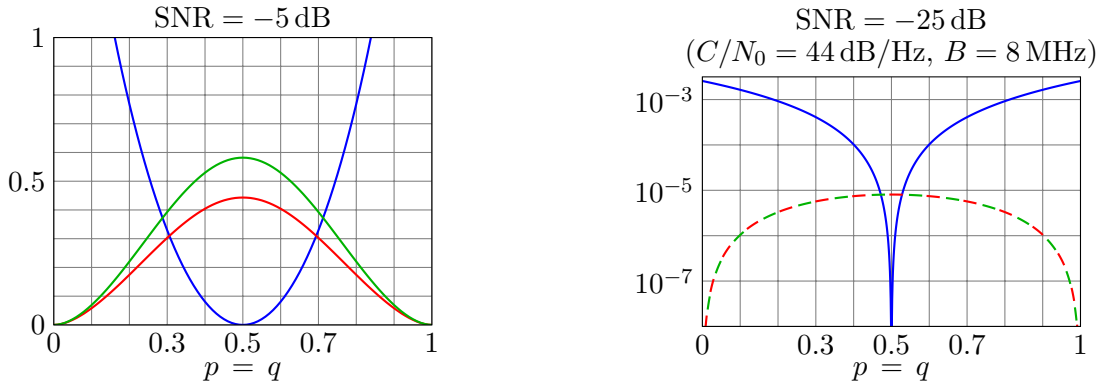
$$= \sum_{m \in \mathcal{A}_\alpha} p_\alpha(m) e^{\dfrac{2am - m^2}{2\sigma^2}} \tag{4.39}$$

the per-chip K-L divergence between the canceled signal and the noise can be computed as:

$$\mathbb{D}(u_i||w_i) = \int f_u(a) \log_2 \frac{f_u(a)}{f_w(a)} da \tag{4.40}$$

$$= \sum_{m \in \mathcal{A}_\alpha} p_\alpha(m) \int f_w(a - m) \log_2 \left[\sum_{n \in \mathcal{A}_\alpha} p_\alpha(n) e^{\dfrac{2am - m^2}{2\sigma^2}} \right] da \tag{4.41}$$

$\mathbb{D}(w_i||u_i)$ can be derived analogously. The results, computed for $p = q$ are shown in Fig. 4.24. As expected, for $p = 0.5$ it is not possible to distinguish among different PRNs, destroying the CDMA concept. On the other end of the range, for $p = 0$ the attacker is able to perfectly cancel the transmitted signal, making it impossible to detect the attack. Thus, the working point shall have $0 < p < 1/2$. Clearly the behavior is symmetric with respect to 0.5.



(a) (b)

Figure 4.24: Per-chip K-L divergence between signals from different SVs (in blue); authentic bit '0' and bit '1' erased by the attacker (in red); and bit '1' erased by the attacker and authentic bit '0' (in green).

The optimal attack strategy is the one that minimizes the per-chip K-L divergence. From Fig. 4.25 it can be seen that the divergence is dependent on the SNR, so the attack strategy should be too.

The optimal attack has $0 \leq q \leq p$ with

$$\begin{cases} q \to 0 & , \text{ for SNR} \to \infty \\ q \to p & , \text{ for SNR} \to 0 \end{cases} \tag{4.42}$$

as shown in Fig. 4.26, where it can be seen that for low SNR the optimal attack strategy is to match the flipping probability of the legitimate signal δ_i. When the SNR increases, the optimal chip-flipping probability of the attacker ε_i decrease and tends to 0. It should be noted that for GNSS signals the usual SNR for LOS signals received with a low gain antenna is around -25dB, and can be much lower for non LOS signals. In this SNR region the attacker is not required to know anything other than the exact position of the victim receiver antenna. On the other hand high gain antennas with 30+ dB gain are available on the market, which move the working point to a region where the attacker should also know the SNR at the receiver antenna.

As the SNR increases, also the ability to distinguish between the noise and a canceled signal increase, giving to the receiver the ability to detect much easily an attack. Based on the divergence for the target receiver' SNR, it is possible to select the chip flipping probability p in order to match the desired performance in terms of false alarm and missed detection probability. Moreover, it is possible to trade the ability to detect an attacker on a per-chip basis in order to reduce the cross-correlation problem, balancing this with a longer observation time.

An implementation issue comes with the signal and modulation requirements. Indeed, in order to maximize the efficiency of the high power amplifier the multiplexed signal shall have constant-envelope [114]. Clearly the pulsed nature of the on/off binary modulation is not offering a constant-envelope, so an optimal modulation scheme that accommodates pulsed signals shall be designed. This is left to future work.

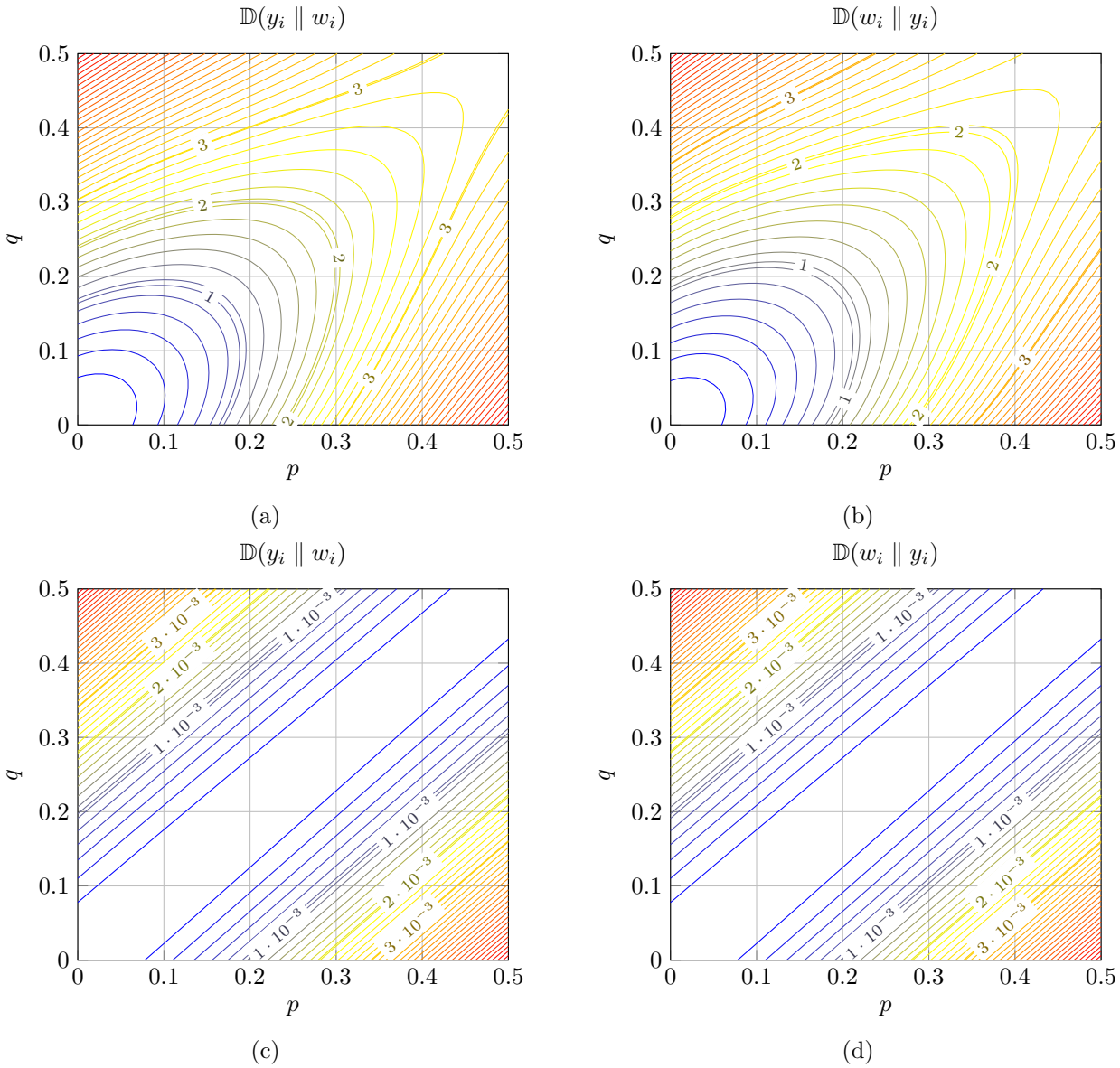


Figure 4.25: Per-chip K-L divergence: in (a),(b) the SNR is set to -5dB , in (c),(d) to -25dB. Figures (a) and (c) represent the divergence between the noise and canceled signal, while (b),(d) between the cancelled signal and noise.

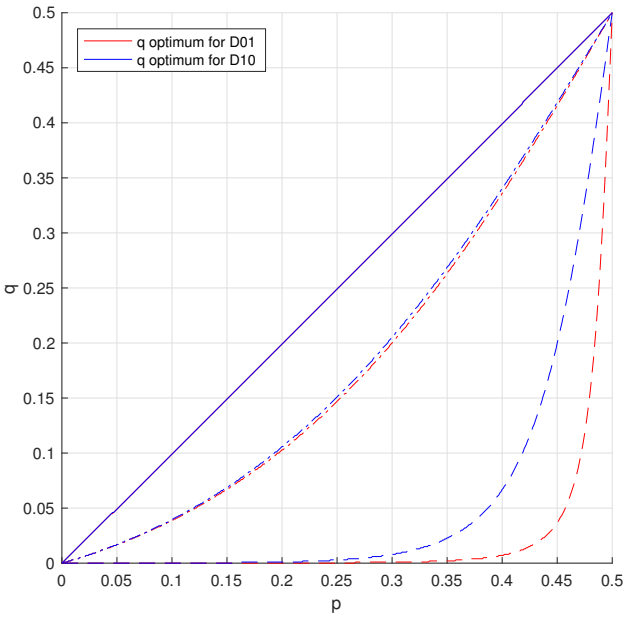


Figure 4.26: Optimal attack strategy against I-code. The dashed line is for SNR = 5 dB, the dotted dashed line for SNR = −5 dB, and the solid line for SNR = −25 dB. The red line is the chip flipping probability that minimize the divergence between the noise and the canceled signal, while in blue line the attack strategy that minimize the divergence between the canceled and the noise.

4.7 Joint data and signal layer authentication

The requirements of a signal layer authentication mechanism can be summarize as:

- shall prevent signal generation attack, it should be hard for an attacker to generate a GNSS like signal that pass the verification and it is used in the PVT computation
- shall provide the capability to distinguish between legitimate signals and intentionally delayed replicas (meaconing, selective delay attacks)

A viable solution to achieve signal layer authentication is the use of SCE. The secret spreading code shall be:

- hard to predict
- reliably reconstructed by a key/seed
- require small bandwidth for the broadcast of the key/seed
- efficiently generated, without requiring computationally intense operations

4.7.1 Detection strategy

Spoofing detection strategy can be classified by where it takes places:

- before correlation, e.g., SCER detection, optimal from the theoretical point of view but requires the knowledge of the attack strategy and of the receiver condition
- after correlation, e.g., detection based on hypothesis testing on the correlation amplitude, are easier to be implemented but achieve suboptimal performance and still requires the knowledge of the attack strategy and of the receiver condition

Many of the detection strategies present in the literature are based on the statistics of the received signals, e.g., SCER see Section 4.4, or on the presence/amplitude of a correlation peak due to the secret spreading code [91, 106]. In Section 4.4 it is shown that correlation amplitude based techniques could be easily influenced by the attacker when applied to NMA. By applying a similar detection strategy to SCE, we can write x as the product of the received signal, $\tilde{s}$, with the reconstructed reliable replica of the secret spreading code, s, after having received and authenticated the key/seed. The pmd of x in the legitimate and spoofed case respectively can be written as:

$$p_{x|l}(a) = \begin{cases} p & a = 1 \\ 1-p & a = -1 \end{cases} \tag{4.43}$$

$$p_{x|s}(a) = \begin{cases} qp + (1-q)(1-p) & a = 1 \\ q(1-p) + p(1-q) & a = -1 \end{cases} \tag{4.44}$$

It is important to note that the same correlation value can be either obtained due to poor SV visibility (low p) or due to highly reliable spoofer estimation (high q). In order to discriminate between the two cases the knowledge of both p and q is needed.

Readers can note that detection strategy based either on the statistics of the received signal or on the correlation amplitude, requires some knowledge of the instantaneous attacker's and receiver conditions.

A different approach is a detection mechanism based on the time of arrival. Similarly to what was discussed in Section 4.5, the unpredictability of the secret spreading code does not allow to generate a

valid signal prior to the transmission of the legitimate signal from the SV. This force the spoofed signal to arrive at the receiver antenna delayed with respect to the legitimate signal. Under the hypothesis that the attacker is not able to block or cancel the legitimate signal, e.g., unplugging the receiver antenna and plugging it to a signal generator, the first identified correlation peak corresponds to the legitimate signal.

4.7.2 Spreading code encryption and NMA integration

Assuming that NMA data are provided by SigAm, see Section 3.5 or TESLA, see Section 3.2.2, the values taken from the one-way chain can be used as a seed for generating the secret spreading code similarly to what proposed in [79]. An advantage of the use of SigAm over TESLA is that it results in a simpler receiver without requiring the implementation of MAC algorithms and that the authentication tags jointly authenticates the navigation message and generates the secret spreading code. To generate the spreading code any stream cipher can be used, e.g., AES or Grain 128a [88]. Being the seed smaller than the produced spreading sequence, it shall be dimensioned to fulfill the desired guessing probability. Depending on the modulation, SCE might be either interleaved with the open spreading code or multiplexed with it. In the former case the secret part shall be allocated over spare symbols, because the receiver will not be able to demodulate the symbols until the reception of the seed; while in the latter approach the secret spreading sequence might be even continuously multiplexed. The interleaving approach could result in some issues for some tracking loop design [115], while the multiplexed approach results in a correlation loss and could increase cross correlation.

Assuming that the secret spreading code is only partially encrypted, e.g., SSSC [91], the receiver can operate on the open part of the spreading code with limited performance degradation. This degradation affects equally all the receivers, independently on whether they are interested or not in authentication. After the acquisition, tracking and data demodulation using the open component, the receiver obtains the navigation message and NMA data. After verifying their authenticity, the authentication tags can be used as seed to generate a local replica of the secret spreading code. During the tracking, the receiver buffers the raw RF samples corresponding to the search window where it is expected the presence of the SCE. At this point the receiver can perform an acquisition using the secret portion of the code on the buffered samples. Three cases can be observed:

- no correlation peak: corresponds to a signal forging attack
- one correlation peak: corresponds either to the normal situation, or corresponds to a meaconing attack or signal blocking (exclude by hypothesis). This can also correspond to a meaconing attack with a longer delay that the observed time windows
- two correaltion peaks: corresponds to a multipath or meaconing/selective delay attack

In order to avoid multiple correlation peaks an opportunity for the attacker is to cancel the legitimate signal. This can be accomplished by generating a replica of the signal that arrives at the receiver antenna exactly the same amplitude of the legitimate signal but with opposite phase. The unpredictability of the signal, guaranteed by the hypothesis on the secret spreading code, forces the attacker to estimate and replay the signal on the fly, mounting a SCER like attack. This attack can be limited by increasing the chipping rate. Moreover, simply estimating the chip is not enough, the attacker shall also know with very high precision the distance between its transmitting antenna and the receiver one, and shall be able to predict the effect of the channel on the transmitted signal (amplitude and phase) [111]. Therefore, mounting a successful signal cancellation attack is highly challenging.

The position of the SCE correlation peak obtained is compared with the one of the open code. If inconsistency are observed, spoofing is declared. The spoofing detection mechanism shall accept all

the signals that shown a single correlation peak or multiple peaks that lie in the expected multipath envelope. This might result in accepting spoofing signal as valid if their impact on the PVT is comparable to that of multipath, while those spoofing attacks that aim at inducing a considerable PVT error will be detected.

The receiver will be able to detect attacks that impose a delay that falls in the search window. Therefore, the definition of the search window is at the same time a degree of freedom of the design, both at the system level (seed disclosure rate), and at the receiver level (storage capacity) and a critical point for the security of the scheme. Similar to the delayed key disclosure for TESLA, a lower seed disclosure rate requires a looser receiver time synchronization. Indeed, an attacker shall obtain a bigger time advantage in order to get the seed from the SIS before generating and transmitting to the victim a valid ranging signal. On the other hand wider search windows requires larger storage capacity and leads to higher computational burden in the SCE acquisition, but allows detecting spoofing attacks with larger ranging difference.

The most critical phase is the cold start acquisition, where the receiver may have a poor time synchronization. It is important to have a good local oscillator that reduces the time uncertainty window at startup. During the acquisition of open code, the receiver shall reject any ranging signal that does not agree with the time uncertainty. If the acquisition can not be performed because for any reason the receiver has a clock error, it shall not update the local clock using GNSS but a secure time transfer, e.g., secure Network Time Protocol (NTP) trough terrestrial data link, is required.

The signal modulation is another system parameter. The SCE signal can be modulated either using the same modulation of the open signal or a specialized one. In the former case the open and the SCE component are subject to the same group delay and mulitpath envelope, thus it is easier to compare the pseudoranges computed on the two component, while in the latter the receiver may not be able to distinguish among different group delay and spoofing with small ranging error. On the other hand signals designing a specialized signal may allow to improve the anti-spoofing capability using higher chipping rate, that are useful to make harder the capture of the tracking loop. Higher chipping rate signals must also be sampled at higher frequency, leading to a higher storage capacity and to more computational intensive operations. A viable solution that lies in between the two approaches may be the use of a CBOC modulation with an open BOC(1,1) component and an encrypted BOC(6,1) component.

4.8 Signal authentication for SBAS

Currently, SBAS is used only for receiving integrity messages, but its signal may also be used for ranging in a similar way to the GNSS one. Thus, at the moment, it is sufficient to protect the SBAS data, as discussed in Section 3.7, but in the future also SBAS ranging protection may be interesting.

In the context of the EAST project, SBAS signal layer authentication was evaluated with the main goal of protecting the SBAS messages. The techniques evaluated are forms of watermarking, where part of the spreading code is overwritten by a cryptographic sequence. One approach is SSSC, see Section 4.2.3, while the second approach does not keep fixed the position of the code overwritten, but a second cryptographic function generates the index of the chips that will be substituted by the secret sequence. It should be noted that, in general, a signal layer verification does not directly imply data authentication. Indeed, SCE protects only the data carried by the encrypted portion of the signal, against an attacker with a limited advantage in terms of antenna gain. Moreover, data authentication through signal level technique is achieved by a hypothesis testing approach and the minimum detectable duration of the attack depends on both the receiver's and attacker' SNR. This means that it is possible that some bit of the SBAS message are altered without failing the authentication verification, in contrast with respect to data level authentication, where a single bit corrupted lead to a failure in the verification. However, for an attacker modify the data while maintaining the consistency

with the signal layer authentication check, it becomes a much more challenging task.

Two approaches are possible:

- *real time authentication,* will require the adoption of security modules, for preventing the user to use the secret cryptographic material to spoof other receivers
- *delayed key disclosure,* will avoid security modules, being suitable for open service

Unfortunately the delayed key disclosure will require a constant data exchange. Data that, in turn, shall be authenticated, thus all the consideration of data layer authentication applies also to signal layer authentication with delayed key disclosure. Moreover, the signal layer authentication require overriding at least part of the public spreading code with a secret spreading code that it is not available to open service users that doesn't have access to the cryptographic material for the verification. This cause an auto-correlation loss and increase the BER and might increase the cross-correlation, degrading the performance of users not interested in authentication.

Therefore, the data layer authentication seems to be the most affordable solution, while the signal layer authentication techniques should be taken into account only if providing ranging through GEO satellites is considered an interesting feature of the system.

Chapter 5

Physical layer authentication for IoT

This Chapter will discuss message authentication though physical layer authentication in a IoT environment. The results on this topic were published in [8, 9].

Message authentication is an important feature of communications systems, and it will become more and more important as multiple devices will autonomously communicate without much user intervention in IoT scenarios. Moreover, in this pervasive context many devices will include very simple capabilities, will communicate in a wireless fashion, and will be significantly limited by energy constraints. Therefore, new authentication schemes that do not require heavy exchange of cryptographic keys and use of security protocols may be extremely useful. Currently, authentication mechanisms are already deployed in Wireless Sensors and Actuators Networks (WSAN) and operate at the Medium Access Control or higher layers using cryptographic approaches: for example, the IEEE 802.15.4 standard encompasses the extension of counter mode encryption and cipher block chaining message authentication code [116] algorithm.

A possible solution to ease authentication is provided by *physical layer authentication* mechanisms, where the features of the channel over which transmission occurs are exploited. As described in the survey paper [117] the physical layer authentication mechanisms can be divided into two categories: one using keys for the authentication and the other not requiring keys. Key-based authentication schemes have been extensively studied in the '80s [118, 119]. In [98] message authentication is interpreted as a hypothesis testing problem, thus extending the previous scenarios. More recently, the presence of noise in the authentication procedure (still based on keys) has been considered in [120, 121]. Recent studies have been focused on joint channel and authentication coding [99]. The problem of key-based authentication schemes, however, is that a shared key is needed, having in turn the problem of generating and managing the key, which can be particularly difficult for non-controlled devices with constrained computation capabilities. Therefore, we focus on key-less authentication that only relies on the characteristics of the channel over which the communication occurs. In particular, it is exploited the fact that the legitimate receiver knows the channel, while the eavesdropper sees another channel due to a different position with respect to the legitimate receiver. In this case, the receiver can perform a two-stage authentication by which it first estimates the channel using a message that has been authenticated by some other means and, for forthcoming messages, it checks if the channel is the same of the first transmission. The attacker can suitably process the transmitted signal in order to let the receiver estimate a different channel, and various deterministic attack strategies have been considered in [122, 100]. A statistical attack strategy has been investigated in [123], while in [124] it has been proved that secure authentication is possible when the legitimate transmitter has a noisy channel to the receiver whose behavior cannot be completely simulated by the attacker.

In this work, it is considered an IoT network where many *source* nodes aim at exchanging messages with a single *concentrator* node, i.e., a Cellular Internet of things (CIoT) network [125, Sec. I]. To this end, they are assisted by *anchor* nodes that are trusted and connected securely with the concentrator

node. Let us define the set of the anchor nodes as *anchor network*. In this context, the aim is to provide a message authentication scheme based on the channel characteristics between the source nodes and the anchor nodes. Assuming that the anchor nodes have a limited energy availability, suitable scheduling policies for the activation of the anchor nodes for authentication purposes to maximize the anchor network lifespan while guaranteeing given False Alarm (FA) and Missed Detection (MD) probabilities of the authentication process will be proposed.

5.1 Physical layer authentication

Let us consider an IoT network with M legitimate sources and N anchor nodes (with indices $i = 1, \dots, N$) and one concentrator node $\mathtt{c}$. In the IoT scenario, where devices transmit at a low rate, we assume that the communication channel between each source $\mathtt{s}$ and each anchor node i is narrowband and can be represented by a single complex coefficient. We suppose that the communication between the anchor nodes and the concentrator node $\mathtt{c}$ is secure, either by some additional communication feature or also because these nodes are connected to node $\mathtt{c}$ by a wired network. The anchor nodes are battery-powered and we assume that for each transmission a fixed amount of energy E_0 is consumed so that each anchor node is able to perform at most Q message authentications.

Let $h_i(\mathtt{s})$ be the channel gain from the generic source node $\mathtt{s}$ to anchor node $i = 1, \dots, N$. $h_i(\mathtt{s})$ includes the effects of path loss, typically deterministic given the node positions, and those of shadowing and fading, modeled as random. The channel power gain is

$$\mathrm{E}[|h_i(\mathtt{s})|^2] = \lambda_i \quad , \qquad i = 1, \dots, N. \tag{5.1}$$

In the following we will assume independent fading over each source-anchor link and for each transmission, thus

$$\mathrm{E}[h_i(\mathtt{s}) h_j^*(\mathtt{s})] = 0 \quad , \qquad \forall i \neq j \tag{5.2}$$

where $\cdot^*$ denotes the complex conjugate operator. Let λ_i be the path loss value of the link between anchor node i and the source node $\mathtt{s}$. We collect the channel gains into the N-size row vector $\boldsymbol{h}(\mathtt{s})$, i.e.,

$$\boldsymbol{h}(\mathtt{s}) = [h_1(\mathtt{s}), \dots, h_N(\mathtt{s})] \, . \tag{5.3}$$

5.1.1 Attacker model

The *attacker node* $\mathtt{a}$ aims at transmitting a message by impersonating a legitimate source node. The attacker is able to both listen to ongoing transmissions and transmit signals. Moreover, we assume that the attacker may be equipped with multiple antennas, in order to be able to beamform signals, conveying messages with different gains to the various anchor nodes.

We indicate by $\boldsymbol{z}$ the random vector of observations available to the attacker, (e.g., the channel gains from $\mathtt{s}$ and $\mathtt{c}$ to $\mathtt{a}$ and/or the channel gains from $\mathtt{a}$ to the N anchor nodes) that we assume to be somewhat correlated with $\boldsymbol{h}(\mathtt{s})$. In particular, if $\boldsymbol{z} = [z_1, \dots, z_N]$ is a vector of N independent observations, we assume that each one is correlated only with the channel from $\mathtt{s}$ to a single anchor node or the concentrator node, with

$$\frac{\mathrm{E}[z_i h_i^*(\mathtt{s})]}{\mathrm{E}[|z_i|^2]} = \rho \, , \quad i = 1, \dots, N \, , \tag{5.4}$$

$$\mathrm{E}[z_i h_j^*(\mathtt{s})] = \mathrm{E}[z_i z_j^*] = 0 \, , \qquad \forall i \neq j. \tag{5.5}$$

Note that (5.4) establishes that the correlation coefficient is the same for all anchor nodes and (5.5) yields that only source-anchor and attacker-anchor channels relative to the same anchor are correlated. However, we will assume that the attacker does neither know vector $\boldsymbol{h}(\mathtt{s})$ nor has access to its estimate.

This is reasonable since the only way to have access to this estimate is to place spoofing nodes very close to each anchor node in order to estimate the same channel. The only knowledge available to the attacker on channel $\boldsymbol{h}(\mathtt{s})$ is its joint statistics with the observations $\boldsymbol{z}$.

We denote by $\boldsymbol{g} = [g_1, \ldots, g_N]$ the vector containing the forged channel gains from the attacker to $\mathtt{c}$ and the N anchor nodes. We also assume that both the source nodes and the concentrator do not know about the presence of the attacker, thus, they do not have an estimate of $\boldsymbol{z}$ or $\boldsymbol{g}$ available.

5.1.2 Authentication protocol

We assume now that all N anchor nodes cooperate for authentication purposes, although in the following of this Chapter the case in which a subset of anchor nodes is active in the authentication process will also be considered . We observe that $\mathtt{c}$ is empowered by the N assisting nodes for the authentication procedure, thus it can be seen $\mathtt{c}$ as a receiver with N distributed antennas. The authentication procedure then operates into two phases.

Phase 1: by some secure pairing procedure it is assumed that the message transmitted in this phase is truly originated by $\mathtt{s}$. The N anchor nodes listen to the transmission and estimate the channel to $\mathtt{s}$ (which by reciprocity is assumed as that from $\mathtt{s}$ to the anchor nodes). Let $\hat{h}_i^{(0)}(\mathtt{s})$ be the channel estimate at anchor node $i = 1, \ldots, N$. This estimate is reported to $\mathtt{c}$.

Phase 2: forthcoming messages transmitted by $\mathtt{s}$ contain an unencrypted header that indicates the message source. Whenever the anchor nodes detect the $\mathtt{s}$ transmission header, they estimate the channel. Upon transmission of the k-th message ($k \geq 1$) with $\mathtt{s}$'s header, let $\hat{h}_i^{(k)}(\mathtt{s})$ be the estimated channel at anchor node i, which is forwarded to $\mathtt{c}$ through an *authentication packet* over a secure channel. We collect all channel estimates relative to packet k into the row vector

$$\hat{\boldsymbol{h}}^{(k)}(\mathtt{s}) = [\hat{h}_1^{(k)}(\mathtt{s}), \ldots, \hat{h}_N^{(k)}(\mathtt{s})] \quad k \geq 1\,. \tag{5.6}$$

$\mathtt{c}$ will take a decision about the message's authenticity on the base of the obtained channel estimates. If the actual transmitter is $\mathtt{s}$, $\hat{h}_i^{(k)}(\mathtt{s})$ will be an estimate of $h_i(\mathtt{s})$, $i = 1, \ldots, N$. Instead, if the actual transmitter is the attacker, since it can induce any channel $\boldsymbol{g}$ to the anchor and concentrator nodes, $\hat{h}_i^{(k)}(\mathtt{s})$ will be an estimate of g_i, $i = 1, \ldots, N$.

The authentication performed by $\mathtt{c}$ on $\hat{\boldsymbol{h}}^{(k)}(\mathtt{s})$ must discern between two hypotheses

- $\mathcal{H}_0$: packet k comes from $\mathtt{s}$,
- $\mathcal{H}_1$: packet k has been transmitted by the attacker $\mathtt{a}$.

The decision between the two hypotheses is taken by comparing estimates $\hat{\boldsymbol{h}}^{(k)}(\mathtt{s})$, $k > 0$ with estimates $\hat{\boldsymbol{h}}^{(0)}(\mathtt{s})$.

In the following we assume that the channel realization in two subsequent phases is subject to different fading (while still correlated), while sources do not move over the phases, so that the path loss for each node remains constant. Once a packet is deemed as authentic, the estimate $\hat{\boldsymbol{h}}^{(0)}(\mathtt{s})$ is updated with the estimate $\hat{\boldsymbol{h}}^{(k)}(\mathtt{s})$ in order to track channel variations over time. Moreover, we assume AWGN.

5.1.3 Decision process

When the transmission is not performed by $\mathtt{s}$, it is expected that the channel estimates of Phase 2 significantly differ from those of Phase 1. However, even when the transmission is actually performed

by $\mathtt{s}$, the estimates in the two phases may differ due to occurred channel variations, noise and interference. Therefore, the decision process is prone to two well known types of errors: a) *False Alarms (FAs)*, occurring when a legitimate packet is deemed as not being transmitted by $\mathtt{s}$, and b) *Missed Detections (MDs)*, occurring when the impersonation attack succeeds, and the message coming from the attacker $\mathtt{a}$ is accepted as authentic. The quality of the detection process is determined by the probabilities of these two events, and in general a lower value of one yields a higher value of the other.

The detection procedure that, for a given FA probability, minimizes the MD probability is the LRT. However, this approach requires the knowledge of the statistics of the channel of the attacker node. Moreover, if $\mathtt{a}$ is able to forge the channels to the anchor nodes, the LRT technique requires the knowledge of the attacking strategy, i.e., vector $\boldsymbol{g}$. Therefore, it is unrealistic to have all this knowledge, and LRT must be dropped in favor of the GLRT [102]. In GLRT the knowledge of $\boldsymbol{g}$ is replaced by its maximum likelihood (ML) estimate, i.e., $\hat{\boldsymbol{h}}^{(k)}(\mathtt{s})$.

This test works as follows. Let $f_{\hat{\boldsymbol{h}}^{(k)}(\mathtt{s})|\mathcal{H}_0}(\mathtt{a})$ be the pdf of $\hat{\boldsymbol{h}}^{(k)}(\mathtt{s})$ under hypothesis $\mathcal{H}_0$. Similarly, let $f_{\hat{\boldsymbol{h}}^{(k)}(\mathtt{s})|\mathcal{H}_1,\boldsymbol{g}}(\mathtt{a}|\boldsymbol{b})$ be the pdf of $\hat{\boldsymbol{h}}^{(k)}(\mathtt{s})$ under hypothesis $\mathcal{H}_1$ and given that $\boldsymbol{g}=\boldsymbol{b}$. Then the LLR of the estimated channel $\hat{\boldsymbol{h}}^{(k)}(\mathtt{s})$ is defined as[1]

$$\log \frac{f_{\hat{\boldsymbol{h}}^{(k)}(\mathtt{s})|\mathcal{H}_1,\boldsymbol{g}}(\hat{\boldsymbol{h}}^{(k)}(\mathtt{s})|\hat{\boldsymbol{h}}^{(k)}(\mathtt{s}))}{f_{\hat{\boldsymbol{h}}^{(k)}|\mathcal{H}_0}(\hat{\boldsymbol{h}}^{(k)}(\mathtt{s}))} \propto \frac{2}{\sigma^2}\|\hat{\boldsymbol{h}}^{(k)}(\mathtt{s})-\hat{\boldsymbol{h}}^{(0)}(\mathtt{s})\|^2 = \Psi\,. \tag{5.7}$$

where $\sigma^2 = \mathrm{E}[\|\hat{\boldsymbol{h}}^{(k)}(\mathtt{s})-\hat{\boldsymbol{h}}^{(0)}(\mathtt{s})\|^2]$. According to the GLRT, the authenticity is established by comparing the LLR (5.7) (or its proportional variable Ψ) with a threshold (ϑ), i.e.,

$$\begin{cases} \Psi \le \vartheta & , \text{ decide for } \mathcal{H}_0 \\ \Psi > \vartheta & , \text{ decide for } \mathcal{H}_1 \end{cases} \tag{5.8}$$

We note from (5.7) that Ψ is a random variable depending both on the estimate accuracy and on the fact that the transmitting node is either $\mathtt{s}$ or $\mathtt{a}$. In particular, conditioned on $\mathcal{H}_0$ and for any realization of $\boldsymbol{h}(\mathtt{s})$, as shown in [100], Ψ is a central chi-square distributed random variable with $2N$ degrees of freedom, yielding the FA probability

$$P_{\mathrm{FA}} = \mathrm{P}[\Psi > \vartheta|\mathcal{H}_0] = 1 - F_{2N,0}(\vartheta)\,, \tag{5.9}$$

where $F_{z,y}(x)$ is the CDF of a non-central chi-square random variable with z degrees of freedom and non-centrality parameter y. On the other hand, conditioned on $\mathcal{H}_0$, specific realizations of $\boldsymbol{h}(\mathtt{s})$ and the forged vector $\boldsymbol{g}$, Ψ is a noncentral chi-square distributed random variable with $2N$ degrees of freedom and noncentrality parameter

$$\beta = \frac{2}{\sigma^2}\|\boldsymbol{g}-\boldsymbol{h}(\mathtt{s})\|^2\,, \tag{5.10}$$

yielding the MD probability

$$P_{\mathrm{MD}}(\boldsymbol{h}(\mathtt{s}),\boldsymbol{g}) = \mathrm{P}[\Psi \le \vartheta|\mathcal{H}_1,\boldsymbol{h}(\mathtt{s}),\boldsymbol{g}] = F_{2N,\beta}(\vartheta)\,. \tag{5.11}$$

We observe that the MD probability depends on the attack channel vector $\boldsymbol{g}$, which is random because it depends on the attacker observations and on its attack strategy. For instance, if Rayleigh fading is assumed and $\boldsymbol{h}(\mathtt{s})$, $\boldsymbol{z}$ jointly circularly symmetric complex Gaussian (CSCG) vectors, the optimal attack – both in the maximum MD probability sense of [100] and in the minimum divergence sense of [123] – is itself jointly CSCG with $\boldsymbol{h}(\mathtt{s})$ and $\boldsymbol{z}$ and can be written as

$$\boldsymbol{g} = \boldsymbol{\Xi}\boldsymbol{z} = \boldsymbol{\Omega}\boldsymbol{h}(\mathtt{s}) + \boldsymbol{\varepsilon} \tag{5.12}$$

[1]We use log for the natural (base-e) logarithm.

with $\boldsymbol{\Xi}$ and $\boldsymbol{\Omega}$ complex matrices, and $\boldsymbol{\varepsilon}$ a zero mean CSCG vector independent of $\boldsymbol{h}(\mathtt{s})$.

Under the assumption of (5.5) both the matrices $\boldsymbol{\Xi}$ and $\boldsymbol{\Omega}$ in (5.12) as well as the covariance matrices of $\boldsymbol{z}$ and $\boldsymbol{\varepsilon}$ are diagonal, see [100, App. A] or [123, Sect. V], so that we can write

$$g_i = \lambda_i z_i = \omega_i h_i(\mathtt{s}) + \varepsilon_i \tag{5.13}$$

where $\omega_i = \lambda_i \rho_{z_i h_i(\mathtt{s})} \sigma_{z_i} / \sigma_{h_i(\mathtt{s})}$ and $\sigma^2_{\varepsilon_i} = |\lambda_i|^2 \sigma^2_{z_i}(1 - |\rho_{z_i h_i(\mathtt{s})}|^2)$, while $\sigma_{h_i(\mathtt{s})}$ and σ_{z_i} represent the standard deviations of $h_i(\mathtt{s})$ and z_i, respectively.

It is worth to assess the average MD probability when the optimal attack is performed and the channel $\boldsymbol{h}(\mathtt{s})$ is Gaussian distributed with i.i.d. entries, i.e., $P_{\mathrm{MD}} = \mathrm{P}[\Psi \leq \vartheta | \mathcal{H}_1]$. This measure is relevant when the sequence of transmitted messages is long enough to span a significant portion of the channel fading statistics[2]. In the case of N independent observations, $\beta = 2\sum_i |(1-\omega_i)h_i(\mathtt{s}) + \varepsilon_i|^2/\sigma^2$ becomes the sum of N independent exponentially distributed random variables, each with mean (see also [100, App. A])

$$\frac{1}{\zeta_i} = \frac{2}{\sigma^2}(|1 - b_i|^2 \lambda_i + \sigma^2_{\varepsilon_i}) = \frac{2}{\sigma^2}(1 - |\rho_{z_i h_i(\mathtt{s})}|^2)\lambda_i \,. \tag{5.14}$$

Then, under the simplifying assumption[3] that the ζ_i are all distinct, the average MD probability is

$$P_{\mathrm{MD}} = 2\sum_{i=1}^{N} \zeta_i \left(\prod_{j \neq i} \frac{1}{1 - \zeta_i/\zeta_j} \right) \left(\sum_{m=0}^{\infty} \frac{\bar{\gamma}(N+m+1; \vartheta/2)}{(2\zeta_i + 1)^{m+1}} \right) \tag{5.15}$$

where $\bar{\gamma}(r; a) = \frac{1}{\Gamma(r)} \int_0^a x^{r-1} e^{-x}\, dx$ denotes the normalized lower incomplete Gamma function.

Observe that, since $F_{2N,x}(\vartheta)$ is a decreasing function of x for every ϑ, and the CDF of β is a decreasing function of each λ_i once $\rho_{z_i h_i(\mathtt{s})} = \rho$ is kept fixed, P_{MD} is itself a decreasing function of each λ_i for a fixed ρ. In other words, better legitimate channel gains yield a lower probability of confusing an attacker as a legitimate source.

5.1.4 Efficient admissible configurations

Until now the case in which all anchor nodes provide the channel gain estimate to the concentrator node $\mathtt{c}$ for each data packet was considered. It is reasonable to assume that most of the energy cost of the anchor nodes comes from their transmission of the authentication packets to the concentrator node $\mathtt{c}$. From (5.15) we observe that nodes having a better channel estimation can better contribute to the authentication as they provide a lower MD probability, once the FA probability and hence the decision threshold, is fixed. On the other hand, in a scenario in which the anchor nodes are battery-powered wireless devices with limited energy storage capabilities, it is important to optimize the use of the anchor nodes. The optimization procedure must on the one hand ensure an accurate message authentication, and on the other hand reduce the power consumption of the anchor nodes.

In the following some definitions that will be useful for the optimization procedure are given.

Configuration

We indicate as *configuration* a set of anchor nodes that are active in authenticating a message presumably coming from source $\mathtt{s}$. Clearly, with N anchor nodes 2^N configurations are possible, and each configuration can be described by a N-size binary vector, where the n-th entry is set to one if anchor node n is active in the authentication process, and zero otherwise. Therefore, the ℓ-th configuration

[2] We remark that fading is independent on each phase, therefore MD is averaged over the fading. The case of constant fading over the phases can be addressed by a similar approach but leads to intractable expressions.

[3] This assumption in only made here for the sake of obtaining a more compact expression in (5.15). In case it does not hold, the pdf of β can be derived with only a slight complication as described in [126].

is that having a vector representation $\boldsymbol{c}_\ell(\mathtt{s})$ being the binary representation of ℓ. For example, for $N = 9$, vector $\boldsymbol{c}_{26}(\mathtt{s}) = [0\,1\,0\,1\,1\,0\,0\,0\,0]^T$ denotes the twenty-sixth configuration for the authentication of node $\mathtt{s}$, where anchor nodes 2, 4 and 5 are active (in red), while anchor nodes 1, 3, 6, 7, 8 and 9 are not active (in blue), as shown in Fig. 5.1. For a selected configuration ℓ, the authentication procedure is carried out as described in the previous section, where the channel vector $\boldsymbol{h}^{(k)}(\mathtt{s})$ has now length

$$L_\ell(\mathtt{s}) = \sum_{i=1}^{N} [\boldsymbol{c}_\ell(\mathtt{s})]_i \,, \tag{5.16}$$

i.e., the Hamming weight of vector $\boldsymbol{c}_\ell(\mathtt{s})$.

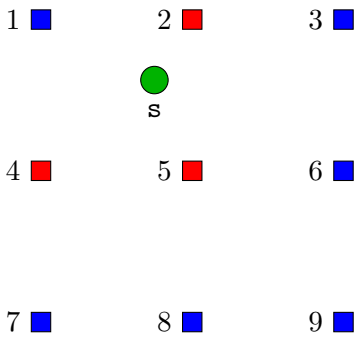


Figure 5.1: Configuration example corresponding to $\boldsymbol{c}_{26}(\mathtt{s}) = [0\,1\,0\,1\,1\,0\,0\,0\,0]^T$, where only the anchor nodes 2, 4 and 5 are active (in red).

Admissible configurations

Not all configurations allow to achieve target FA and MD probabilities, therefore these configurations must be discarded in the optimization process. The configurations that provide at most the target MD and FA probabilities, will be denoted as *admissible configurations*. In particular, configuration $\boldsymbol{c}_\ell(\mathtt{s})$ is admissible if exists a ϑ such that both the constraints on FA and MD probabilities, adapted to the configuration $\boldsymbol{c}_\ell(\mathtt{s})$, are satisfied, that is

$$P_{\text{FA}} = 1 - F_{2L_\ell(\mathtt{s}),0}(\vartheta) \,, \tag{5.17}$$

and

$$P_{\text{MD}} = 2 \sum_{i=1}^{N} [\boldsymbol{c}_\ell(\mathtt{s})]_i \zeta_i \left[\prod_{j \neq i} \frac{1}{(1 - \zeta_i/\zeta_j)^{[\boldsymbol{c}_\ell(\mathtt{s})]_i}} \right] \sum_{m=0}^{\infty} \frac{\bar{\gamma}(L_\ell(\mathtt{s}) + m; \vartheta/2)}{(2\zeta_i + 1)^{m+1}} \tag{5.18}$$

are not higher than their respective target values.

Efficient configuration

Since in the following we aim at selecting the admissible configurations that yield longer network lifespan, which is related to the usage of anchor nodes in the network, we observe that if a configuration $\boldsymbol{c}_\ell(\mathtt{s})$ is admissible in which anchor node i is not active ($[\boldsymbol{c}_\ell(\mathtt{s})]_i = 0$), the configuration $\boldsymbol{c}'_\ell(\mathtt{s})$ obtained by activating node i ($[\boldsymbol{c}'_\ell(\mathtt{s})]_i = 1$, and $[\boldsymbol{c}'_\ell(\mathtt{s})]_j = [\boldsymbol{c}_\ell(\mathtt{s})]_j \ \forall q \neq i$) yields additional power consumption while still being admissible. Therefore, the newly obtained configuration $\boldsymbol{c}'_\ell(\mathtt{s})$ is worse than the original $\boldsymbol{c}_\ell(\mathtt{s})$ in terms of energy consumption. On these grounds, only *efficient* admissible configurations those with a minimal set of active nodes is considered.

Let $a_{\mathtt{s}}$ be the number of efficient admissible configurations for source node $\mathtt{s}$, and denote each configuration as $\boldsymbol{c}_\ell(\mathtt{s})$, $\ell = 1 \dots a_{\mathtt{s}}$. We can collect all efficient admissible configurations into the $N \times A$ binary matrix

$$\boldsymbol{C} = [\ \boldsymbol{c}_1(1)\ \cdots\ \boldsymbol{c}_{a_1}(1)\ \cdots\ \boldsymbol{c}_1(M)\ \cdots\ \boldsymbol{c}_{a_M}(M)\ ]\ , \tag{5.19}$$

where

$$A = \sum_{m=1}^{M} a_m \tag{5.20}$$

is the total number of efficient admissible configurations.

5.2 Network lifespan

As a metric to assess the performance of the proposed methods, we consider the *anchor network lifespan*, defined as the smallest number of authentication processes after which at least one anchor node runs out of power. The underlying assumption is that if any anchor node is no longer available, there will be some source position for which no efficient admissible configuration exists, therefore, no reliable authentication can be performed.

Since the choice of the configuration is random, the network lifespan is a random variable, too. In order to derive its statistical description, let us denote with $y_i(k)$, $i = 1, \dots, N$, $k \geq 1$, the number of message authentications performed by anchor node i out of the first k authentications performed by the network. Recalling that Q denotes the maximum number authentication process to which an anchor node can take part due to the finite battery capacity, the CDF of the random lifespan L of the anchor network can be written as

$$F_L(k) \triangleq \mathbb{P}[L \leq k] = \mathbb{P}[\max_i y_i(k) > Q]\,. \tag{5.21}$$

The evaluation of the above expression requires the joint distribution of $y_i(k)$, which is fairly complicated by the correlations introduced by the specific set of efficient admissible configurations. However, denoting with u_i the probability of using anchor node n, one can easily see that the marginal distribution of each $y_i(k)$ is binomial with parameters (k, u_i). Then, we can upper and lower bound the CDF in (5.21) as

$$\max_i \mathbb{P}[y_i(k) > Q] \leq \mathbb{P}[\max_i y_i(k) > Q] \leq \sum_i \mathbb{P}[y_i(k) > Q] \tag{5.22}$$

and hence

$$I_{\max_i u_i}(Q+1, k-Q) \leq F_L(k) \leq \sum_i I_{u_i}(Q+1, k-Q)\,, \tag{5.23}$$

where $I_x(a,b) \triangleq B(x;a,b)/B(1,a,b)$ is the regularized incomplete beta function, $B(x;a,b) = \int_0^x t^{a-1}(1-t)^{b-1}dt$.

Observe that the term on the left is the CDF of a *negative binomial* random variable, so that an upper bound on the expected network lifespan can be obtained by integrating the Complementary CDF (CCDF) of its upper bound as

$$\begin{aligned} \mathbb{E}[L] &= \sum_k [1 - F_L(k)] \\ &\leq \sum_k [1 - I_{\max_i u_i}(Q+1, k-Q)] \\ &= \frac{Q+1}{\max_i u_i}\,, \end{aligned} \tag{5.24}$$

where the last equality is given by the mean of the negative binomial random variable.

However, the bounds in (5.23)–(5.24) may be rather loose, as will be seen by the numerical results in Section 5.4, so we will also resort to the approximation of $F_L(k)$ that can be obtained by neglecting the statistical dependence among $y_1(k), \ldots, y_N(k)$, that is

$$\begin{aligned} F_L(k) &\simeq 1 - \prod_{n=1}^{N} \mathbb{P}[y_i(k) > Q] \\ &= 1 - \prod_{n=1}^{N} \left[1 - I_{u_i}(Q+1, k-Q)\right]. \end{aligned} \tag{5.25}$$

Although not justified, the above approximation is seen to be quite good from the numerical results in Section 5.4.

5.3 Anchor Node Selection Criteria

Since the anchor devices have a limited energy budget, the choice of the configuration used for authentication becomes critical for the ability of the CIoT to continue performing the authentication operations. The optimization process will select, among the admissible configurations, those that will minimize the average energy consumption of the most used anchor node. Instead of selecting a single configuration and always using it, we allow for a mixed utilization of configurations, where each configuration is used for a fraction of the times in which a message coming (presumably) from source `s` must be authenticated. In practice, this mixed use of configurations can be implemented by picking at random the configuration to be used according to specific probability distribution. Therefore, the optimization process aims at selecting the usage probability distribution in order to maximize the anchor network lifespan.

Let $p_\ell(m)$ be the probability (or the fraction times) of using the configuration $\boldsymbol{c}_\ell(m)$ and let us stack the probability mass function (PMF) of configurations into the A-size column vector

$$\boldsymbol{\pi} = \left[\; p_1(1) \quad \cdots \quad p_{a_1}(1) \quad p_1(2) \quad \cdots \quad p_1(M) \quad \cdots \quad p_{a_M}(M) \;\right]^T \tag{5.26}$$

where $\cdot^T$ denotes the transpose operator. Define $\boldsymbol{u}$ as the N-size column vector having as entry n the probability that anchor node n is used for authentication (*anchor utilization probability*), irrespective of the used source node. Assume that φ_m is the probability that source node $m \in \{1, \ldots, M\}$ is transmitting, therefore $\sum_{m=1}^{M} \varphi_m = 1$. Let us define the $A \times A$ diagonal matrix $\boldsymbol{\Phi}$ that weights the admissible configurations by the probabilities that the corresponding transmitter is active, i.e.,

$$\boldsymbol{\Phi} = \begin{bmatrix} \varphi_1 \cdot \boldsymbol{I}_{a_1} & 0 & 0 & 0 \\ 0 & \varphi_2 \cdot \boldsymbol{I}_{a_2} & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \varphi_M \cdot \boldsymbol{I}_{a_M} \end{bmatrix} \tag{5.27}$$

where $\boldsymbol{I}_n$ is the identity matrix of size $n \times n$. Then $\boldsymbol{u}$ can be written as

$$\boldsymbol{u} = \boldsymbol{C}\boldsymbol{\Phi}\boldsymbol{\pi}\,. \tag{5.28}$$

In most cases it is reasonable to assume that each source is transmitting with the same probability $\varphi_m = 1/M \;\forall m$, and in this case we have

$$\boldsymbol{u} = \frac{1}{M}\boldsymbol{C}\boldsymbol{\pi}\,. \tag{5.29}$$

5.3.1 Upper bound maximization

Since the upper bound on the expected lifespan (5.24) is inversely proportional to the maximum value of u_i, a first approach for the optimization of $\boldsymbol{\pi}$ is the minimization of $\max_i u_i$, under the constraint that only efficient admissible configurations are used each time. The optimization problem can then be written as follows:

$$\min_{\boldsymbol{\pi}} \max_i u_i \tag{5.30a}$$

subject to (5.26), (5.28) and

$$0 \leq \pi_\ell(\mathtt{s}) \leq 1\,, \quad \ell = 1, \ldots, a_{\mathtt{s}}\,,\ \mathtt{s} = 1, \ldots, M\,, \tag{5.30b}$$

$$\sum_{\ell=1}^{a_{\mathtt{s}}} \pi_\ell(\mathtt{s}) = 1\,, \quad \mathtt{s} = 1, \ldots, M\,. \tag{5.30c}$$

We remark that the constraint (5.30c) ensures that for each source node to be authenticated there is always a configuration to be used when requested. If by taking off some node there still exists an efficient admissible configuration, the optimization problem can be easily fixed by ignoring the index of that node in the maximization.

The min-max problem (5.30) can be solved as a linear programming problem

$$\min_{\boldsymbol{\pi}, t} t \tag{5.31a}$$

subject to (5.26), (5.28), (5.30b), (5.30c) and

$$\frac{1}{M}\boldsymbol{C}\boldsymbol{\pi} \leq t\mathbf{1}_{N\times 1}\,. \tag{5.31b}$$

where $\mathbf{1}_{N\times 1}$ is an N-size column vector with entries all equal to 1.

We must observe that by solving the min-max problem we are maximizing an upper bound on the average anchor network lifetime, or its complementary CDF; however, maximizing the bound does not necessary correspond to maximize the bounded value (average or complementary CDF). Therefore, we explore two other possible methods to choose the configuration probability, based on the minimization of the variance of u_i or on the minimization of their power, respectively.

5.3.2 Minimum variance optimization

By solving the min-max problem for the relevant scenario described in the next section, we have noticed that in most cases the optimized node utilization u_i are almost constant, i.e., $u_i \approx u_j\ \forall i, j$. This is also intuitive if one thinks that, starting from a feasible solution, the utilization of the most used node can be reduced by increasing the probability of efficient admissible configurations that do not contain that node, thus increasing the utilization of other anchor nodes. Therefore, let us choose $\boldsymbol{\pi}$ in order to minimize the variance of u_i

$$f(\boldsymbol{\pi}) = \sum_{i=1}^{N} \left(u_i - \frac{1}{N}\sum_{j=1}^{N} u_j \right)^2 . \tag{5.32}$$

Denoting by $\mathbf{1}_{N\times N}$ the $N \times N$ matrix containing all entries equal to 1, $f(\boldsymbol{\pi})$ can be expressed in matrix form as follows:

$$\begin{aligned} f(\boldsymbol{\pi}) &= ||\boldsymbol{C}\boldsymbol{\pi} - \frac{1}{N}\mathbf{1}_{N\times N}\boldsymbol{C}\boldsymbol{\pi}||^2 \\ &= ||\boldsymbol{C}\boldsymbol{A}\boldsymbol{\pi}||^2 \\ &= \boldsymbol{\pi}^T\boldsymbol{C}^T\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{C}\boldsymbol{\pi} \\ &= \boldsymbol{\pi}^T\boldsymbol{C}^T\boldsymbol{A}\boldsymbol{C}\boldsymbol{\pi}\,, \end{aligned} \tag{5.33}$$

where $\boldsymbol{A} \triangleq \boldsymbol{I}_N - \frac{1}{N}\mathbf{1}_{N\times N}$ with $\boldsymbol{I}_N$ the $N \times N$ identity matrix, is a symmetric and idempotent matrix.

The problem of minimizing (5.32) can now be written as

$$\min_{\boldsymbol{\pi}} \boldsymbol{\pi}^T \boldsymbol{C}^T \boldsymbol{A} \boldsymbol{C} \boldsymbol{\pi} \tag{5.34}$$

subject to (5.26), (5.28), (5.30b), and (5.30c). Note that the objective function of problem (5.34) is convex and constraints (5.26), (5.28), (5.30b), and (5.30c) are affine transformations. The optimization problem is convex and can be solved using well-known techniques such as the interior point method.

5.3.3 Least squares optimization

The minimum variance optimization aims at making all utilization probabilities similar to each other, however, do not explicitly minimize the average node utilization probabilities. Therefore, as a third optimization method is considered the minimization of the sum of the square probabilities of utilization across the anchor nodes, i.e.,

$$\min_{\boldsymbol{\pi}} \sum_i u_i^2 = \min_{\boldsymbol{\pi}} \boldsymbol{\pi}^T \boldsymbol{C}^T \boldsymbol{C} \boldsymbol{\pi} \tag{5.35}$$

subject to (5.26), (5.28), (5.30b), and (5.30c). Also in this case the convexity of the objective function and the affine nature of the constraints make the problem easily solvable.

5.4 Numerical Results

In the following we will reference to a CIoT scenario, where the IoT is deployed in the Global System for Mobile communications (GSM) frequency bands, which has been standardized by the 3rd generation partnership project (3GPP) [125]. Let us consider a CIoT for a campus, extending over a circular area with a radius of 500 m. The anchor nodes and the concentrator node are placed on a square grid inside the circle; the number of anchor nodes is in the set $\{4, 9, 16, 25, 36\}$. We assume a unitary transmit power. The deterministic component of the wireless channel, i.e., the path loss, is computed as (in dB)

$$(\Gamma(\eta, d))_{\text{dB}} \triangleq -10 \cdot \eta \cdot \log_{10}\left(\frac{4\pi d}{\Lambda}\right), \tag{5.36}$$

where η is the Path Loss Exponent (PLE), d is the distance between the transmitter and the receiver, Λ is the wavelength, defined as the ratio between the speed of light and the carrier frequency f. We assume that $f = 900$ MHz, which is the typical carrier frequency value considered in the context of CIoT. The parameter η is in the interval $[2, 3]$, which is a reasonable assumption for the radio-wave propagation in an urban scenario. We recall that as the Path Loss Exponent (PLE) η increases, the propagation environment becomes harsher.

5.4.1 Missed detection probability

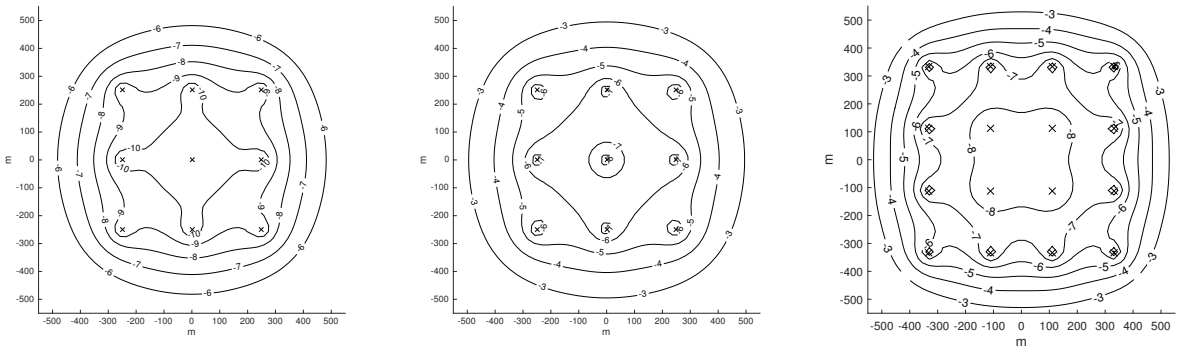
The performance will be assessed choosing the detection threshold that ensures a FA probability $P_{\text{FA}} = 10^{-4}$.

Fig. 5.2 shows (in log scale) the average (with respect to noise and channel realization) MD probability as a function of the legitimate source node position, with $N = 9$ anchor nodes and a correlation factor for the attacker node $\rho \in \{0.1, 0.5\}$ in Fig. 5.2a.-Fig. 5.2b and $N = 16$ anchor nodes and a correlation factor for the attacker node $\rho = 0.2$ in Fig. 5.2c. The PLE is set to $\eta = 2$ in Fig. 5.2a and Fig. 5.2b, while $\eta = 2.5$ in Fig. 5.2c. The SNR, defined as the average (over fading) power ratio for a sensor-anchor distance of 250 meters, is 15 dB. We observe that the positions at the center of the circle provide a lower MD probability, since the average channel gain sensed by the anchor nodes is

Parameter	Value
f	900 MHz
η	2
ρ	0.1
Cell radius	500 m
SNR	30 dB (at 250 m)
N	9
M	10

Table 5.1: Simulation parameters for the anchor network lifespan evaluation.

higher than for external positions, especially as the source node moves to the circle border. The same scenario is considered for Fig. 5.3, that report the CCDF of the MD probability for different values of N and PLE, with $SNR = 8$ dB at 250 m and $\rho = 0.1$. It can easily be seen that an increasing value of η impacts the performance of the proposed authentication protocol much more than an increasing number of anchor nodes N.



(a) (b) (c)

Figure 5.2: Logarithm of the MD probability as a function of legitimate source node position for (a) $\rho = 0.1$ and (b) $\rho = 0.5$, $N = 9$, $SNR = 15$ dB at a distance of 250 m; (c) $N = 16$ anchor nodes, and a correlation factor $\rho = 0.2$. The PLE is set to $\eta = 2$ in (a)-(b) and to $\eta = 2.5$ in (c).

5.4.2 Anchor network lifespan

The performance of the proposed authentication method and the various approaches for the choice of the configuration probabilities will now be assessed in terms of the anchor network lifespan. The system parameters used in the simulation are summarized in Table 5.1.

Fig. 5.4 shows the CDF of the network lifespan L for the min-max, the minimum variance and the minimum power methods of Section 5.3. We report the CDF of the empirical lifespan obtained by Monte Carlo simulation, together with the upper and lower bound of the CDF (see (5.23)–(5.24)). Moreover, the approximation of the CDF obtained by assuming independent anchor node usage (see (5.25)) is reported, indicated with label "independent" in the figures. It can be observed that the lower bound has a quite loose performance, while both the independent approximation and the upper bound are quite close to the empirical CDF. When comparing the various optimization methods, we observe that they perform approximately the same. In order to better assess the differences among the methods, Fig. 5.5 reports the empirical CDF for the various optimization methods. We observe that

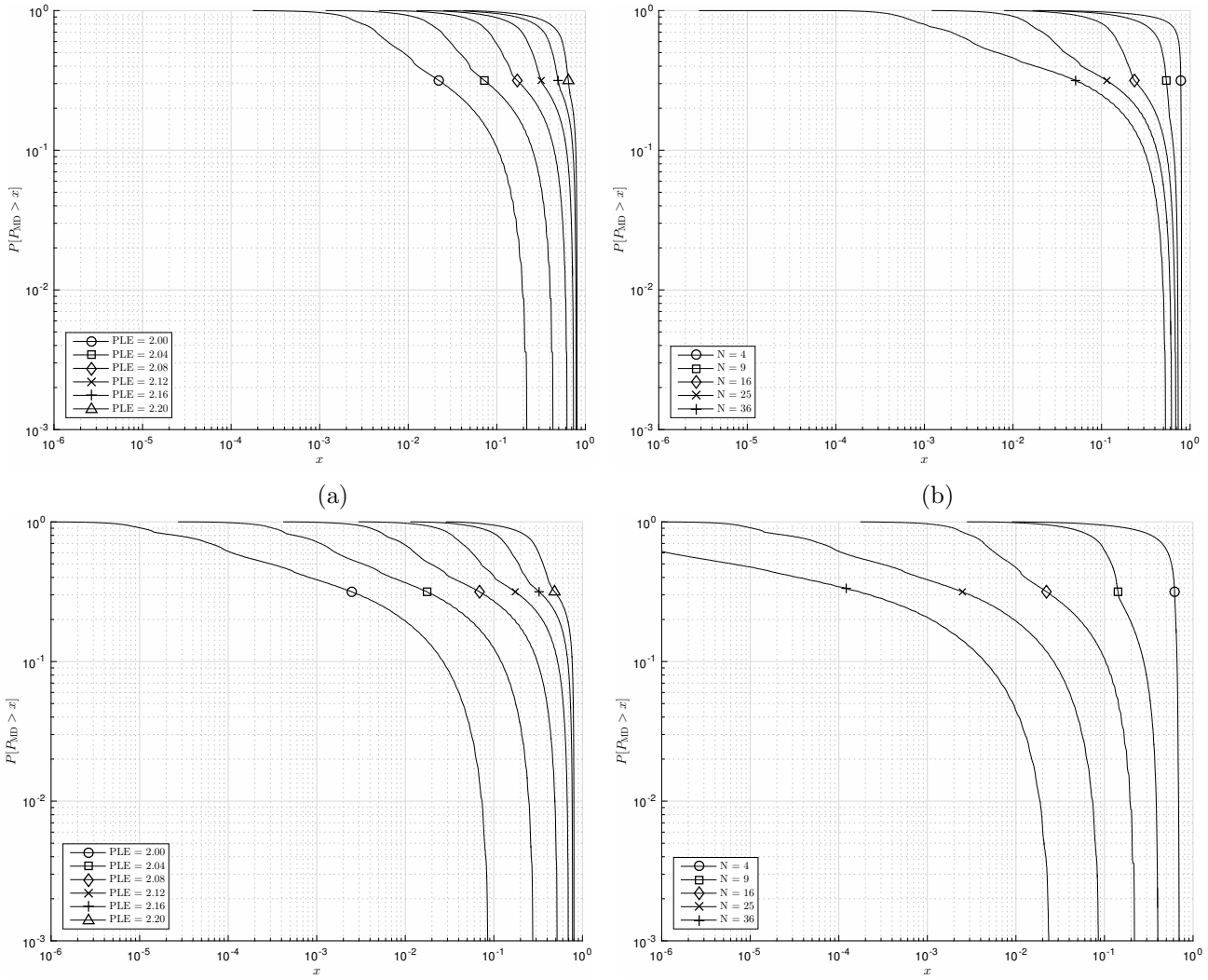
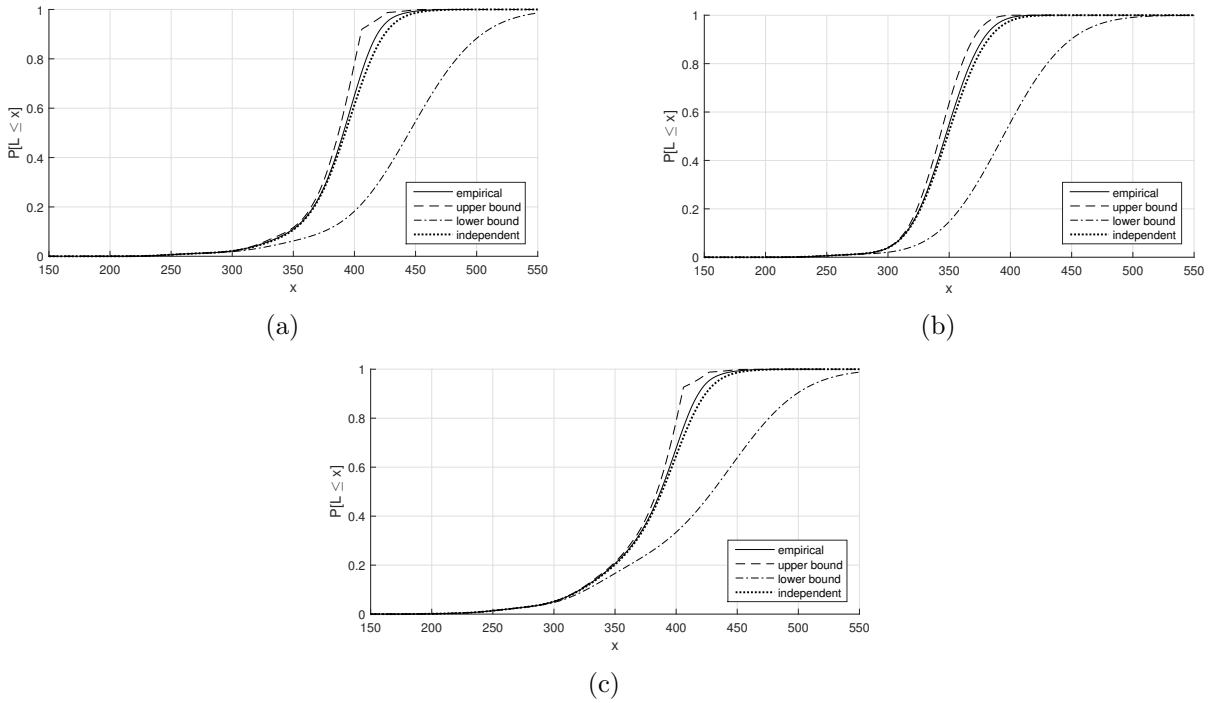


Figure 5.3: CCDF of MD probability in the case of fixed number of anchor nodes and PLE, with $SNR = 8$ dB at 250 m, $\rho = 0.1$ and $N = 16$ in (a), $N = 25$ in (c) and PLE = 2.1 in (b), PLE = 2.0 (d).

the min-max optimization provides the highest anchor network lifespan, while the minimum variance and minimum power approaches have different behaviors at different outage probabilities.



(a) (b)

(c)

Figure 5.4: Empirical CDF and bounds of the anchor network lifespan L using the configuration probability vector $\boldsymbol{\pi}$ obtained solving the min-max problem (a), the minimum variance problem (b), and the least squares problem (c).

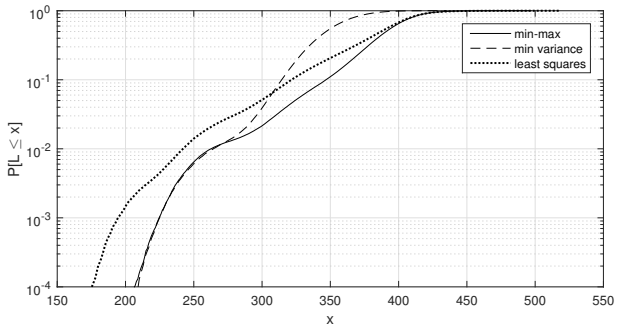


Figure 5.5: Empirical CDF of the anchor network lifespan for the various optimization methods.

Chapter 6

Conclusions and Recommendations

6.1 Conclusions

This chapter concludes the thesis presenting the conclusions from the discussed research activity. This thesis focused on the authentication and cryptographical integrity protection for critical infrastructures. The focus was on the Global Navigation Satellite Systems (GNSS) and Internet of Things (IoT). The security of the satellite navigation was analyzed at different levels. At the data level the current authentication schemes presented in the literature were analyzed both from the security and the performance point of view. An outcome of these analyses is that one-way chain computed using iterative padding and truncation of a secure hashing function are not ideal and allow brute force attacks that leverage the increasing collision probability with the number of steps attacked. From the lessons learned by these analyses a novel authentication scheme, SigAm, was designed to be cryptographically secure and to achieve good performance. Moreover, the scheme was designed to be complemented with a signal level mechanism that aims at protecting the ranging. Satellite-Based Augmentation System (SBAS) authentication was taken into account, identifying the more suitable data layer authentication schemes for the context.

At the signal level, the SCER estimation strategy was generalized, allowing the selection of the attack parameters in order to minimize the detection probability. At the same time, it was shown that the optimal detection strategy is based on the complete LRT detection. The latter requires the knowledge of the attack parameters, that is a non realistic assumption, thus the GLRT detection strategy was introduced. The reported experimental analysis on SCER detection revealed that this attack is a concrete threat, being easy to mount and hard to detect. Alternatives mechanism at signal level was developed: an autonomous anti-spoofing mechanism that exploits semi-codeless tracking and the encrypted military service was proposed, and an adaptation of the I-code modulation to the GNSS context was made.

In the IoT context a physical layer authentication mechanism was evaluated. Due to the resource constrained devices, different node activation polices were evaluated, in order to minimize the energy consumption and maximize the network lifespan, defining an optimal energy efficient strategy.

6.2 Recommendations for future work

In order to achieve a resilient PNT service, the data layer authentication of GNSS signal is not enough. Future research activity should integrate the design of data and signal layer authentication mechanism in a single authentication scheme. This can be performed on the existing ranging signals or, alternatively, a dedicated authentication component can be designed. In the latter case, the system designer could have more freedom in the design, jointly designing not only the authentication scheme but also other aspects such as the modulation and the FEC scheme.

Beside the activity on the system side, the development of optimal receiver-based techniques that exploit the authentication features provided by the SIS, will be needed. With the increase of complexity of the receivers, these receiver-based mechanisms might also include input from other sensors or systems, allowing authenticated hybrid position.

Bibliography

[1] J. A. Volpe, "Vulnerability Assessment of the Transportation Infrastructure Relying on the Global Positioning System," tech. rep., National Transportation Systems Center, 2001.

[2] U. Kröner, H. Greidanus, R. Gallagher, M. Sironi, G. Azzalin, F. Littmann, P. Tebaldi, P. Timossi, and D. Shaw, *Report on Authentication in Fisheries Monitoring.* 2009.

[3] G. Caparra, S. Sturaro, N. Laurenti, and C. Wullems, "Evaluating the security of one-way key chains in TESLA-based GNSS Navigation Message Authentication schemes," in *2016 International Conference on Localization and GNSS (ICL-GNSS)*, (Barcellona), pp. 1–6, IEEE, jun 2016.

[4] G. Caparra, S. Sturaro, N. Laurenti, C. Wullems, and R. T. Ioannides, "A Novel Navigation Message Authentication Scheme for GNSS Open Service," in *ION GNSS+ 2016*, (Portland, Oregon), pp. 2938 – 2947, 2016.

[5] G. Caparra, C. Wullems, S. Ceccato, S. Sturaro, N. Laurenti, O. Pozzobon, R. T. Ioannides, and M. Crisci, "Design Drivers for Navigation Message Authentication Schemes for GNSS Systems," *InsideGNSS*, vol. 11, no. 5, pp. 64–73, 2016.

[6] G. Caparra, N. Laurenti, R. T. Ioannides, and M. Crisci, "Improving Secure Code Estimation and Replay Attack and Detection on GNSS Signals," in *ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing, NAVITEC*, 2014.

[7] G. Caparra, C. Wullems, and R. T. Ioannides, "An Autonomous GNSS Anti-Spoofing Technique," in *Navitec 2016*, (Noordwijk, The Netherlands), 2016.

[8] G. Caparra, M. Centenaro, N. Laurenti, S. Tomasin, and L. Vangelista, "Energy-based anchor node selection for IoT physical layer authentication," in *2016 IEEE International Conference on Communications (ICC)*, (Kuala Lumpur, Malaysia), pp. 1–6, IEEE, may 2016.

[9] G. Caparra, M. Centenaro, N. Laurenti, S. Tomasin, and L. Vangelista, "Energy-Efficient Physical Layer Authentication in the Internet of Things," in *Information Theoretic Security and Privacy of Information Systems* (R. F. Schaefer, H. Boche, A. Khisti, and H. V. Poor, eds.), Cambridge University Press, 2017.

[10] Navipedia, "Correlators - Navipedia," in *http://www.navipedia.net/index.php/Correlators*, (Date Accessed: 2016-11-15), 2011.

[11] E. D. Kaplan and C. J. Hegarty, *Understanding GPS: principles and applications.* Artech house, 2005.

[12] M. Appel, A. Konovaltsev, and M. Meurer, "Joint Antenna Array Attitude Tracking and Spoofing Detection Based on Phase Difference Measurements," in *ION GNSS+ 2016*, (Portland, Oregon), 2016.

[13] K. D. Wesson, D. P. Shepard, J. A. Bhatti, and T. E. Humphreys, "An Evaluation of the Vestigial Signal Defense for Civil GPS Anti-Spoofing," *Proceedings of the 24th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS 2011)*, pp. 177–193, 2011.

[14] D. M. Akos, "Who's Afraid of the Spoofer? GPS/GNSS Spoofing Detection via Automatic Gain Control (AGC)," *Navigation*, vol. 59, pp. 281–290, dec 2012.

[15] A. Jafarnia Jahromi, *GNSS Signal Authenticity Verification in the Presence of Structural Interference*. PhD thesis, University of Calgary, 2013.

[16] K. D. Wesson, M. Rothlisberger, and T. E. Humphreys, "Practical Cryptographic Civil GPS Signal Authentication," *Journal of the Institute of Navigation*, pp. 1–15, 2011.

[17] C. Yang, M. Miller, E. Blasch, and T. Nguyen, "Comparative study of coherent, non-coherent, and semi-coherent integration schemes for GNSS receivers," in *Proceedings of the 63rd Annual Meeting of the Institute of Navigation*, pp. 572–588, 2001.

[18] R. T. Ioannides, L. E. Aguado, and G. Brodin, "Diverse Signals Combinations for High-Sensitivity GNSS," *Journal of Navigation*, vol. 60, p. 497, sep 2007.

[19] R. van Nee, J. Siereveld, P. Fenton, and B. Townsend, "The multipath estimating delay lock loop: approaching theoretical accuracy limits," in *Proceedings of 1994 IEEE Position, Location and Navigation Symposium - PLANS'94*, pp. 246–251, IEEE, 1994.

[20] G. Gamba, S. Fantinato, O. Pozzobon, M. Anghileri, R. T. Ioannides, and J.-Á. Ávila-Rodríguez, "The Spoofing Estimating Delay Lock Loop," in *Navitec 2014*, (Noordwijk, The Netherlands), 2014.

[21] A. J. Jahromi, A. Broumandan, S. Daneshmand, G. Lachapelle, and R. T. Ioannides, "Galileo signal authenticity verification using signal quality monitoring methods," in *2016 International Conference on Localization and GNSS (ICL-GNSS)*, pp. 1–8, IEEE, jun 2016.

[22] C. O'Driscoll, J.-Á. Ávila-Rodríguez, and R. T. Ioannides, "Bandlimiting and Dispersive Effects on High Order BOC Signals," in *ION GNSS+ 2016*, (Portland, Oregon), 2016.

[23] S. Hewitson and J. Wang, "GNSS Receiver Autonomous Integrity Monitoring (RAIM) Performance Analysis," *GPS Solutions*, vol. 10, no. 3, pp. 155–170, 2006.

[24] J. Ventura-Traveset and D. Flament, *EGNOS – The European Geostationary Navigation Overlay System*. ESA, sp-1303 ed., 2006.

[25] ICAO, *Annex 10 - Volume 1 Aeronautical Telecommunications - Radio Navigation Aids*.

[26] R. T. Ioannides, T. Pany, and G. Gibbons, "Known Vulnerabilities of Global Navigation Satellite Systems, Status, and Potential Mitigation Techniques," *Proceedings of the IEEE*, vol. 104, pp. 1174–1194, jun 2016.

[27] N. O. Tippenhauer, C. Pöpper, K. B. Rasmussen, and S. Capkun, "On the requirements for successful GPS spoofing attacks," in *ACM conference on Computer and communications security, CCS*, (New York, New York, USA), p. 75, ACM Press, 2011.

[28] A. Pando and D. Horacio, "Distance-Decreasing Attack in Global Navigation Satellite System," 2009.

[29] K. Zhang and P. Papadimitratos, "GNSS receiver tracking performance analysis under distance-decreasing attacks," in *IEEE International Conference on Location and GNSS, ICL-GNSS*, pp. 1–6, IEEE, jun 2015.

[30] T. E. Humphreys, "Detection Strategy for Cryptographic GNSS Anti-Spoofing," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, pp. 1073–1090, apr 2013.

[31] O. Pozzobon, C. Wullems, and M. Detratti, "Security considerations in the design of tamper resistant GNSS receivers," in *ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing, NAVITEC*, pp. 1–5, IEEE, dec 2010.

[32] C. Wullems, O. Pozzobon, and K. Kubik, "Signal Authentication and Integrity Schemes for Next Generation Global Navigation Satellite Systems," in *European Navigation Conference, (ENC-GNSS)*, pp. 1–10, 2005.

[33] A. Dalla Chiara, G. Da Broi, O. Pozzobon, S. Sturaro, G. Caparra, N. Laurenti, J. Fidalgo, M. Odriozola, J. Caro Ramon, I. Fernández-Hernández, and E. Chatre, "Authentication Concepts for Satellite-Based Augmentation Systems," in *ION GNSS+ 2016*, (Portland, Oregon), pp. 3208 – 3221, 2016.

[34] A. Perrig and J. Tygar, "Secure Broadcast Communication," *Wired and Wireless Networks, Springer*, 2003.

[35] K. Grover and A. Lim, "A survey of broadcast authentication schemes for wireless networks," *Ad Hoc Networks*, vol. 24, pp. 288–316, jan 2015.

[36] N. P. Smart, V. Rijmen, B. Gierlichs, K. G. Paterson, M. Stam, B. Warinschi, and G. Watson, "Algorithms, Key Size and Parameters Report," tech. rep., ENISA, 2014.

[37] J. Jonsson and B. Kaliski, "Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1," 2003.

[38] "ISO/IEC 9796-2:2010. Information technology — Security techniques — Digital signature schemes giving message recovery — Part 2: Integer factorization based mechanisms," ISO ISO/IEC\char9796-2:2010, International Organization for Standardization, Geneva, Switzerland, 2010.

[39] D. Pointcheval and S. Vaudenay, "On provable security for digital signature algorithms," tech. rep., 1996.

[40] C.-P. Schnorr, "Efficient Identification and Signatures for Smart Cards," in *Proceedings of the 9th Annual International Cryptology Conference on Advances in Cryptology*, CRYPTO '89, (London, UK, UK), pp. 239–252, Springer-Verlag, 1990.

[41] S. C. Lo and P. K. Enge, "Authenticating aviation augmentation system broadcasts," in *IEEE/ION Position, Location and Navigation Symposium*, pp. 708–717, IEEE, may 2010.

[42] A. J. Kerns, K. D. Wesson, and T. E. Humphreys, "A blueprint for civil GPS navigation message authentication," in *2014 IEEE/ION Position, Location and Navigation Symposium - PLANS 2014*, pp. 262–269, IEEE, may 2014.

[43] G. Ateniese and B. de Medeiros, "A Provably Secure Nyberg-Rueppel Signature Variant with Applications," *IACR Cryptology ePrint Archive*, 2004.

[44] D. Boneh, B. Lynn, and H. Shacham, "Short Signatures from the Weil Pairing," in *Advances in Cryptology — ASIACRYPT 2001*, pp. 514–532, 2001.

[45] A. Perrig, "The BiBa one-time signature and broadcast authentication protocol," in *Proceedings of the 8th ACM conference on Computer and Communications Security - CCS '01*, (New York, New York, USA), p. 28, ACM Press, nov 2001.

[46] L. Reyzin and N. Reyzin, "Better than BiBa: Short One-Time Signatures with Fast Signing and Verifying," in *Information Security and Privacy, 7th Australasian Conference, ACISP 2002* (L. Batten and J. Seberry, eds.), (Melbourne, Australia), pp. 144–153, Springer-Verlag Heidelberg, 2002.

[47] A. A. Yavuz, "ETA: efficient and tiny and authentication for heterogeneous wireless systems," in *ACM conference on Wireless network security, WiSec*, pp. 67–72, 2013.

[48] J. T. Curran, M. Paonni, and J. Bishop, "Securing the Open-Service: A Candidate Navigation Message Authentication Scheme for Galileo E1 OS," in *European Navigation Conference, (ENC-GNSS)*, (Rotterdam), 2014.

[49] I. Fernández-Hernández, V. Rijmen, G. Seco-Granados, J. Simón, I. Rodríguez, and J. D. Calle, "Design Drivers, Solutions and Robustness Assessment of Navigation Message Authentication for the Galileo Open Service," in *International Technical Meeting of The Satellite Division of the Institute of Navigation, ION GNSS*, pp. 2810–2827, 2014.

[50] P. Walker, V. Rijmen, I. Fernández-Hernández, G. Seco-Granados, J. Simón, J. D. Calle, and O. Pozzobon, "Galileo Open Service Authentication : A Complete Service Design and Provision Analysis," in *ION GNSS+ 2015*, (Tampa, Florida), 2015.

[51] I. Fernández-hernández, G. Seco-granados, I. Rodríguez, and J. D. Calle, "A Navigation Message Authentication Proposal for the Galileo Open Service," *NAVIGATION: Journal of The Institute of Navigation*, vol. 63, no. 1, pp. 85–102, 2016.

[52] ETSI, *White Paper Quantum Safe Cryptography and Security; An introduction , benefits , enablers and challenges.* 2014.

[53] L. Chen, S. Jordan, Y.-K. Liu, D. Moody, R. Peralta, R. Perlner, and D. Smith-Tone, "Quantum Cryptography Report on Post - Quantum Cryptography," pp. 1–15, 2016.

[54] D. J. Bernstein, J. Buchmann, and E. Dahmen, *Post-Quantum Cryptography.* Berlin, Heidelberg: Springer Berlin Heidelberg, 2009.

[55] N. Courtois, M. Finiasz, and N. Sendrier, "How to achieve a McEliece-based digital signature scheme," in *Advances in Cryptology - ASIACRYPT 2001* (C. Boyd, ed.), (Gold Coast, Australia), pp. 157–174, Springer Berlin Heidelberg, 2001.

[56] M. Finiasz, "Parallel-CFS," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6544 LNCS, pp. 159–170, 2011.

[57] M. Baldi, M. Bianchi, F. Chiaraluce, J. Rosenthal, and D. Schipani, "Using LDGM Codes and Sparse Syndromes to Achieve Digital Signatures," in *Post-Quantum Cryptography, PQ*, pp. 1–15, Springer Berlin Heidelberg, may 2013.

[58] A. Kipnis, J. Patarin, and L. Goubin, "Unbalanced Oil and Vinegar Signature Schemes," in *Advances in Cryptology — EUROCRYPT '99* (J. Stern, ed.), pp. 206–222, Prague, Czech Republic: Springer Berlin Heidelberg, 1999.

[59] J. Ding and D. Schmidt, "Rainbow, a New Multivariable Polynomial Signature Scheme," in *Applied Cryptography and Network Security*, pp. 164–175, New York, USA: Springer Berlin Heidelberg, 2005.

[60] J. Ding, B.-Y. Yang, C.-H. O. Chen, M.-S. Chen, and C.-M. Cheng, "New Differential-Algebraic Attacks and Reparametrization of Rainbow," in *Applied Cryptography and Network Security*, vol. 5037 LNCS, pp. 242–257, Berlin, Heidelberg: Springer Berlin Heidelberg, 2008.

[61] A. Petzoldt, S. Bulygin, and J. Buchmann, "Selecting Parameters for the Rainbow Signature Scheme," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6061 LNCS, pp. 218–240, 2010.

[62] A. Perrig, R. Canetti, J. Tygar, and D. Song, "Efficient authentication and signing of multicast streams over lossy channels," in *Proceeding 2000 IEEE Symposium on Security and Privacy. S&P 2000*, pp. 56–73, IEEE Comput. Soc, 2000.

[63] I. Fernández-Hernández, "Method and system to optimise the authentication of radionavigation signals," *Patent EP14163902*, 2015.

[64] A. Perrig, R. Canetti, J. Tygar, and B. Briscoe, "Timed Efficient Stream Loss-Tolerant Authentication (TESLA): Multicast Source Authentication Transform Introduction", RFC 4082," 2005.

[65] M. Archer, "Proving correctness of the basic TESLA multicast stream authentication protocol with TAME," in *WITS '02*, (Portland, Oregon), 2002.

[66] A. Lomuscio and B. Woźna, "A complete and decidable security-specialised logic and its application to the TESLA protocol," in *Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems - AAMAS '06*, (Hakodate, Hokkaido, Japan), ACM, 2006.

[67] I. Ouranos, K. Ogata, and P. Stefaneas, "Formal Analysis of TESLA Protocol in the Timed OTS/CafeOBJ Method," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7610 LNCS, pp. 126–142, 2012.

[68] J. Guilford, K. Yap, and V. Gopal, "Fast SHA-256 Implementations on Intel Architecture Processors," *IA Architects*, 2012.

[69] Bitmaintech, "https://www.bitmaintech.com/product.htm."

[70] Antpool, "https://www.antpool.com."

[71] Blockchain.info, "https://blockchain.info/charts/hash-rate?timespan=all."

[72] I. Fernández-Hernández, I. Rodríguez, G. Tobías, J. D. Calle, E. Carbonell, G. Seco-Granados, J. Simón, and R. Blasi, "Galileo Commercial Service. Testing GNSS High Accuracy and Authentication," *InsideGNSS*, vol. 10, no. 1, pp. 38–48, 2015.

[73] M. E. Hellman, "A Cryptanalytic Time-Memory Trade-Off," *IEEE Transactions on Information Theory*, vol. 26, no. 4, pp. 401–406, 1980.

[74] J. T. Curran and C. O'Driscoll, "Message Authentication and Channel Coding," in *ION GNSS+ 2016*, (Portland, Oregon), 2016.

[75] J. T. Curran, M. Navarro, M. Anghileri, P. Closas, and S. Pfletschinger, "Coding Aspects of Secure GNSS Receivers," *Proceedings of the IEEE*, vol. 104, pp. 1271–1287, jun 2016.

[76] J. T. Curran, M. Bavaro, P. Closas, M. Anghileri, M. Navarro, B. Schotsch, and S. Pfletschinger, "On the Threat of Systematic Jamming of GNSS," in *ION GNSS+ 2016*, (Portland, Oregon), 2016.

[77] D. Borio, C. Gioia, G. Baldini, and J. Fortuny, "GNSS Receiver Fingerprinting for Security-Enhanced Applications," in *ION GNSS+ 2016*, (Portland, Oregon), 2016.

[78] Y. Liu, J. Li, and M. Guizani, "PKC Based Broadcast Authentication using Signature Amortization for WSNs," *IEEE Transactions on Wireless Communications*, vol. 11, pp. 2106–2115, jun 2012.

[79] J. T. Curran and M. Paonni, "Securing GNSS: An End-to-End Feasibility Study for the Galileo Open Service," in *International Technical Meeting of the Satellite Division of The Institute of Navigation, ION GNSS*, pp. 1–15, 2014.

[80] M. Canale, S. Fantinato, and O. Pozzobon, "Performance Comparison of Different Data Authentication Solutions for the Galileo CS," in *ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing, NAVITEC*, 2014.

[81] N. Zivic and M. F. Flanagan, "On Joint Cryptographic Verification and Channel Decoding via the Maximum Likelihood Criterion," *IEEE Communications Letters*, vol. 16, pp. 717–719, may 2012.

[82] N. Zivic, "Reliability of Soft Verification of Message Authentication Codes," in *2013 Fifth International Conference on Computational Intelligence, Communication Systems and Networks*, pp. 191–196, IEEE, jun 2013.

[83] N. Zivic, "Security Aspects of Soft Verified Messages Protected by Message Authentication Codes," in *2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications*, pp. 1316–1322, IEEE, jun 2012.

[84] T. Walter, J. Blanch, R. Eric Phelts, and P. Enge, "Evolving WAAS to serve L1/L5 users," *Navigation, Journal of the Institute of Navigation*, vol. 59, no. 4, pp. 317–327, 2012.

[85] P. Enge and T. Walter, "Digital Message Authentication for SBAS (and APNT)," 2014.

[86] I. Fernández-Hernández, J. Simón, R. Blasi, C. Payne, T. Miquel, and J. P. Boyero, "The Galileo Commercial Service: Current Status and Prospects," in *European Navigation Conference, (ENC-GNSS)*, (Rotterdam), 2014.

[87] I. Rodríguez, G. Tobías, J. D. Calle, J. M. Martín, O. Pozzobon, M. Canale, D. Maharaj, P. Walker, E. Göhler, P. Toor, and I. Fernández-Hernández, "Preparing for the Galileo Commercial Service – Proof of Concept and Demonstrator Development," in *International Technical Meeting of the Satellite Division of the Institute of Navigation, ION GNSS*, pp. 1–12, 2014.

[88] M. Ågren, M. Hell, T. Johansson, and W. Meier, "Grain-128a: a new version of Grain-128 with optional authentication," *International Journal of Wireless and Mobile Computing*, vol. 5, p. 48, dec 2011.

[89] H. Zhang and X. Wang, "Cryptanalysis of Stream Cipher Grain Family.," *IACR Cryptology ePrint Archive*, vol. 2009, 2009.

[90] M. G. Kuhn, "An asymmetric security mechanism for navigation signals," in *International Workshop on Information Hiding, IH* (J. Fridrich, ed.), vol. 3200 of *Lecture Notes in Computer Science*, (Berlin, Heidelberg), pp. 239–252, Springer Berlin Heidelberg, 2005.

[91] L. Scott, "Anti-spoofing & authenticated signal architectures for civil navigation systems," *Proceedings of the Institute of Navigation GPS/GNSS 2003 conference*, pp. 1543–1552, 2003.

[92] O. Pozzobon, L. Canzian, M. Danieletto, and A. Dalla Chiara, "Anti-spoofing and open GNSS signal authentication with signal authentication sequences," in *ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing, NAVITEC*, pp. 1–6, IEEE, dec 2010.

[93] O. Pozzobon, C. Sarto, A. Pozzobon, D. Dötterböck, B. Eissfeller, E. Pérez, and D. Abia, "Open GNSS signal authentication based on the Galileo Commercial Service (CS)," in *International Technical Meeting of The Satellite Division of the Institute of Navigation, ION GNSS+*, pp. 1–10, 2013.

[94] M. L. Psiaki, B. W. O'Hanlon, J. A. Bhatti, D. P. Shepard, and T. E. Humphreys, "GPS Spoofing Detection via Dual-Receiver Correlation of Military Signals," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, pp. 2250–2267, oct 2013.

[95] T. Bull, "A new high performance way of detecting and mitigating the Jamming Meaconing and spoofing of commercial GNSS signals," in *ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing, NAVITEC*, pp. 1–5, IEEE, dec 2010.

[96] J. Hall, M. Barbeau, and E. Kranakis, "Detection of transient in radio frequency fingerprinting using signal phase," *Wireless and Optical Communications*, pp. 13–18, 2003.

[97] V. Brik, S. Banerjee, M. Gruteser, and S. Oh, "Wireless device identification with radiometric signatures," in *Proceedings of the 14th ACM international conference on Mobile computing and networking - MobiCom '08*, (New York, New York, USA), p. 116, ACM Press, 2008.

[98] U. M. Maurer, "Authentication theory and hypothesis testing," *IEEE Transactions on Information Theory*, vol. 46, pp. 1350–1356, jul 2000.

[99] L. Lai, H. El Gamal, and H. V. Poor, "Authentication Over Noisy Channels," *IEEE Transactions on Information Theory*, vol. 55, pp. 906–916, feb 2009.

[100] P. Baracca, N. Laurenti, and S. Tomasin, "Physical Layer Authentication over MIMO Fading Wiretap Channels," *IEEE Transactions on Wireless Communications*, vol. 11, pp. 2564–2573, jul 2012.

[101] O. Pozzobon, G. Gamba, M. Canale, and S. Fantinato, "Supersonic GNSS Authentication Codes," in *International Technical Meeting of The Satellite Division of the Institute of Navigation, ION GNSS*, pp. 1–8, 2014.

[102] S. M. Kay, *Fundamentals of Statistical Signal Processing, Volume 2: Detection Theory*. New Jersey: Prentice-Hall Inc, 1993.

[103] O. Zeitouni, J. Ziv, and N. Merhav, "When is the generalized likelihood ratio test optimal?," *IEEE Transactions on Information Theory*, vol. 38, no. 5, pp. 1597–1602, 1992.

[104] Greatscottgadgets, "https://greatscottgadgets.com/hackrf/."

[105] T. E. Humphreys, J. A. Bhatti, D. P. Shepard, and K. D. Wesson, "The Texas Spoofing Test Battery : Toward a Standard for Evaluating GPS Signal Authentication Techniques," in *ION GNSS 2012*, (Nashville, Tennessee), pp. 3569 – 3583, 2012.

[106] I. Fernández-Hernández and G. Seco-Granados, "Galileo NMA Signal Unpredictability and Anti-Replay Protection," in *ICL-GNSS 2016*, 2016.

[107] H. V. Poor and O. Hadjiliadis, *Quickest Detection*. Cambridge University Press, 2009.

[108] D. Egea-Roca, G. Seco-Granados, J. A. Lopez-Salcedo, E. Domíngez, L. E. Aguado, D. Lowe, D. Naberzhnykh, F. Dovis, I. Fernández-Hernández, and J. P. Boyero, "Signal-level Integrity and Metrics Based on the Application of Quickest Detection Theory to Interference Detection," in *ION GNSS+ 2015*, (Tampa, Florida), pp. 3136 – 3147, 2015.

[109] Z. Zhang, M. Trinkle, L. Qian, and H. Li, "Quickest detection of GPS spoofing attack," in *MILCOM 2012 - 2012 IEEE Military Communications Conference*, pp. 1–6, IEEE, oct 2012.

[110] K. T. Woo, "Optimum Semi-Codeless Carrier Phase Tracking of L2," in *Navigation*, vol. 47, pp. 82–99, 1999.

[111] M. Cagalj, S. Capkun, R. Rengaswamy, I. Tsigkogiannis, M. Srivastava, and J.-P. Hubaux, "Integrity (I) codes: message integrity protection and authentication over insecure channels," in *2006 IEEE Symposium on Security and Privacy (S&P'06)*, pp. 15 pp.–294, IEEE, 2006.

[112] K. B. Rasmussen, S. Capkun, and M. Cagalj, "SecNav: Secure Broadcast Localization and Time Synchronization in Wireless Networks," in *Proceedings of the 13th annual ACM international conference on Mobile computing and networking - MobiCom '07*, (New York, New York, USA), p. 310, ACM Press, sep 2007.

[113] M. Bertier, A.-M. Kermarrec, and G. Tan, "Message-Efficient Byzantine Fault-Tolerant Broadcast in a Multi-hop Wireless Sensor Network," in *2010 IEEE 30th International Conference on Distributed Computing Systems*, pp. 408–417, IEEE, 2010.

[114] J.-Á. Ávila-Rodríguez, *On Generalized Signal Waveforms for Satellite Navigation*. PhD thesis, Universität der Bundeswehr München, 2008.

[115] O. Pozzobon, "Keeping the Spoofs Out. Signal Authentication Services for Future GNSS," *InsideGNSS*, vol. 6, no. 3, pp. 48–55, 2011.

[116] NIST, "Pub 800-38C, Recommendation for Block Cipher Modes of Operation—The CCM Mode for Authentication and Confidentiality.," tech. rep., U.S. Department of Commerce/N.I.S.T., 2004.

[117] E. Jorswieck, S. Tomasin, and A. Sezgin, "Broadcasting Into the Uncertainty: Authentication and Confidentiality by Physical-Layer Processing," *Proceedings of the IEEE*, vol. 103, no. 10, pp. 1702–1724, 2015.

[118] G. J. Simmons, "Authentication theory/coding theory," in *Advances in Cryptology: Proceedings of CRYPTO 84* (G. R. Blakley and D. Chaum, eds.), pp. 411–431, Springer Berlin Heidelberg, 1985.

[119] G. J. Simmons, "A survey of information authentication," *Proceedings of the IEEE*, vol. 76, no. 5, pp. 603–620, 1988.

[120] P. L. Yu, J. S. Baras, and B. M. Sadler, "Physical-Layer Authentication," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 1, pp. 38–51, 2008.

[121] P. L. Yu, J. S. Baras, and B. M. Sadler, "Power allocation tradeoffs in multicarrier authentication systems," in *IEEE Sarnoff Symp.*, pp. 1–5, 2009.

[122] P. Baracca, N. Laurenti, and S. Tomasin, "Physical Layer Authentication over an OFDM Fading Wiretap Channel," in *Proceedings of the 5th International ICST Conference on Performance Evaluation Methodologies and Tools*, VALUETOOLS '11, (ICST, Brussels, Belgium, Belgium), pp. 648–657, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2011.

[123] A. Ferrante, N. Laurenti, C. Masiero, M. Pavon, and S. Tomasin, "On the Error Region for Channel Estimation-Based Physical Layer Authentication Over Rayleigh Fading," *IEEE Transactions on Information Forensics and Security*, vol. 10, pp. 941–952, may 2015.

[124] S. Jiang, "Keyless Authentication in a Noisy Model," *IEEE Transactions on Information Forensics and Security*, vol. 9, pp. 1024–1033, jun 2014.

[125] A. Biral, M. Centenaro, A. Zanella, L. Vangelista, and M. Zorzi, "The challenges of M2M massive access in wireless cellular networks," *Digital Communications and Networks*, vol. 1, no. 1, pp. 1–19, 2015.

[126] H. V. Khuong and H. Y. Kong, "General expression for pdf of a sum of independent exponential random variables," *IEEE Communications Letters*, vol. 10, no. 3, pp. 159–161, 2006.

Acknowledgments

This work was supported in part by the Advanced GNSS Open Service Signal Integrity Protection and Authentication at the Physical Layer (A GOSSIP A PLAY) activity of the European Space Agency, by the EGNOS Authentication Security Testbed (EAST) activity of the European Commission and by the MIUR project ESCAPADE (Grant RBFR105NLC) under the "FIRB-Futuro in Ricerca 2010" funding program.

I would like to thank all the people that collaborated with me on these topics. I want to start from my supervisor Nicola Laurenti, that I admire for his passion for the research and for his precision on the work. Then I want to thank all the persons form the Radio Navigation Systems and Techniques Section (TEC-ETN), ESA-ESTEC especially the Head of Section, Massimo Crisci, and Rigas T. Ioannides, Christian Wullems and James T. Curran for sharing their experience with me and guiding me throughout the activity. The master students I supervised during their thesis, notably Silvia Sturaro and Silvia Ceccato, contributed to the activity and motivated me with their enthusiasm. I had a fruitful collaboration with Marco Centenaro, Stefano Tomasin and Lorenzo Vangelista that introduced me to the IoT context. A notable support has come from Qascom, in particular from the founders Oscar Pozzobon. I want to thank the reviewer of this thesis, Christina Pöpper and James T. Curran, that with their careful and critic reading contributed to improve the final result.

Lastly, I would like to thank my parents Nicoletta and Cataldo, my brother Mattia, and Giulia that supported me in all my pursuits.