Physical resilience to insider attacks in IoT networks: Independent cryptographically secure sequences for DSSS anti-jamming

Renzo Efrain Navas, Frédéric Cuppens, Nora Boulahia Cuppens, Laurent Toutain, Georgios Papadopoulos

Physical resilience to insider attacks in IoT networks: Independent cryptographically secure sequences for DSSS anti-jamming

Renzo E. Navas[a,*], Frédéric Cuppens[b], Nora Boulahia Cuppens[b], Laurent Toutain[c] and Georgios Z. Papadopoulos[c]

[a]*Lab-STICC, UMR CNRS 6285, F-35700, IMT Atlantique, Rennes, France*
[b]*École Polytechnique of Montréal, Canada*
[c]*IRISA, UMR CNRS 6074, F-35700, IMT Atlantique, Rennes, France*

ARTICLE INFO

Keywords:
Internet of Things
Moving Target Defense
Anti-Jamming
Direct-Sequence Spread-Spectrum
Cryptographically Secure Pseudo-Random
Cross-Correlation

ABSTRACT

Wireless communication is a key technology for the Internet of Things (IoT). Due to its open nature, the physical layer of wireless systems is a high-priority target for an adversary whose goal is to disrupt the normal behavior of the system. In particular, jamming attacks are one of the most straightforward and effective types of attacks: information flow of the system is stopped or severely disturbed. In this paper, we propose a method to improve the jamming resilience of IoT systems based on Direct-Sequence Spread-Spectrum (DSSS) techniques. Our proposal is inspired by the Moving Target Defense (MTD) paradigm. MTD strategies randomize components of a system, increasing the effort an attacker needs to compromise the system. We use state-of-the-art Cryptographically Secure Pseudo-Random Number Generators outputs as spreading sequences for DSSS. The sequences of the proposed system are generated in an ad-hoc, independent, and distributed way. We show probabilistically that the generated sequences have robust cross-correlation properties. We define a multi-user system model to evaluate the Bit-Error-Rate of our proposal in the presence of two types of jammers: a classical band-limited Gaussian noise jammer, and an insider smart jammer with knowledge of one spreading sequence used in the system. Our proposal proactively mitigates the insider jammer attack. We quantify the insider smart jammer resilience of a system implementing our proposal, as a function of the length of the spreading sequences and the jammer power.

1. Introduction

The Internet of Things (IoT) is a reality: every year billions of new heterogeneous devices are capable of interacting through the Internet [1]. Resource-constrained IoT devices account for the majority of these connected devices. Constraints include limited energy, memory, or processing power, among others [2]. This constrained nature makes them attractive from an economic point of view. The low cost of a single device allows for IoT-node networks of large size, where every node has sensors and actuators that interact with our physical world. This creates opportunities for new services like Industry 4.0, Smart Farming, Smart Cities, or Smart Home that will be otherwise financially unviable. However, the sustained increase in connected IoT devices also enables opportunities for new types of cyber-attacks that specifically target them [3, 4]. Moreover, legacy security solutions and protocols are, in general, not applicable to IoT systems. For example, a cell-battery-powered IoT node in a few-bytes-per-day limited network, will not be able to use the WWW-ubiquitous Transport Layer Security protocol in a Public Key Infrastructure-setting. This constrained nature of IoT systems that enables their proliferation, at the same time imposes new yet-unresolved challenges to guarantee even the most basic security objectives. The state-of-the-art in IoT security [5] is lagging behind a reality where billions of vulnerable IoT devices are deployed every year.

A promising cyber-defense approach that can help to change this situation is the Moving Target Defense (MTD) paradigm [6]. MTD tries to disrupt the information asymmetry between a *static* system and unknown attackers, by making components of the target system *inherently dynamic*. The *time* dimension is now a constraint on the attackers because the target system is -ideally- in a perpetual change to unknown states. This MTD dynamism can be applied to one or many components of a system; i.e., data, software, runtime environment, hardware platform, or network layer. While MTD has been gaining growing attention in the last decade [7, 8, 9, 10], MTD research targeted at IoT systems is still limited.

Although a secure system involves security mechanisms at many of its components, the network layer is fundamental. More precisely, the *network physical* (PHY) layer is arguably the most important resource of an IoT system to be protected. First, the PHY layer is the enabler of the distributed capabilities of IoT systems; upper-layer services rely on it. Second, most IoT networks are wireless, and the open nature of this medium makes the PHY layer an easily-accessible *target resource* for an attacker. Among the existing PHY layer attacks, *jamming* is one of the most basic and effective.

A *jammer* introduces a signal into a shared medium to disturb legit communication between nodes in the system. The consequence of a successful jamming attack is that the

*Corresponding author
renzo.navas@imt-atlantique.fr (R.E. Navas);
frederic.cuppens@polymtl.ca (F. Cuppens);
nora.boulahia-cuppens@polymtl.ca (N. Boulahia Cuppens);
laurent.toutain@imt-atlantique.fr (L. Toutain);
georgios.papadopoulos@imt-atlantique.fr (G.Z. Papadopoulos)
ORCID(s): 0000-0002-8784-7444 (R.E. Navas); 0000-0002-0331-0579 (G.Z. Papadopoulos)

information flow of the system is disrupted. Even more, as a consequence of Shannon's limit on any communication channel [11], an attacker with enough power will *always* be successful in jamming a target system with limited power. Therefore, using resources into a jamming attack is an effective strategy from an attacker's point of view [12]. Correspondingly, from a system perspective Anti-Jamming (AJ) defense mechanisms should be a priority.

Spread-spectrum techniques are well-known for their AJ capabilities [13, 14]. Two prominent spread-spectrum techniques are Frequency-Hopping Spread-Spectrum (FHSS), and Direct-Sequence Spread-Spectrum (DSSS). Both techniques rely on a pre-shared sequence between transmitter and receiver to (de)spread the signal in the time-frequency domains. A jammer without the knowledge of the pre-shared sequence cannot power-efficiently jam the transmission. State-of-the-art IoT radio uses spread-spectrum techniques. IEEE 802.15-.4 uses DSSS, and defines a Time Slotted Channel Hopping (TSCH) mode based on FHSS [15]. Long-range LoRa modulation uses patented Chirp Spread-Spectrum (CSS) [16]. However, none of those spread-spectrum systems were designed with AJ as a primary objective. On the one hand, LoRa uses well-known spreading-parameters to do CSS, allowing for trivial jamming [17]. On the other hand, 802.15.4 in DSSS mode uses spreading-sequences not only fixed but also too short for providing any AJ guarantee. Even if 802.-15.4-TSCH provides an FHSS framework which allows a system to use a custom or standardized *hopping schedule*, most of 802.15.4-TSCH networks in the literature can be jammed [18].

There is a lack of IoT systems designed with AJ as one of their primary objectives. Furthermore, insider-node jamming attacks are a real threat to heterogeneous-IoT systems. A malicious insider-node has knowledge of the public network parameters of the system. Thus, it can efficiently jam nodes that share the same AJ parameters.

In this paper, we propose a novel IoT-oriented AJ mechanism inspired by the MTD paradigm and leveraging on spread-spectrum techniques. By design, our proposal proactively mitigates insider-node jamming attacks. Our proposal randomizes the spreading-sequences used by the nodes in a DS-SS system. Every pair of communicating nodes will have a unique pairwise spreading-sequence, only known by them. The novelty of our proposal relies on two factors. First, the spreading-sequences are generated using Cryptographically Secure Pseudo-Random (CSPR) number generators; thus, cryptographically strong randomness claims of the generated sequences are assured. Second, the generation process is done following a decentralized and independent process.

In summary, the main contributions of our work are:

- We propose an IoT-oriented AJ MTD network physical layer strategy that uses CSPR number generators for the randomization of DSSS spreading-sequences.
- We study the cross-correlation statistical and probabilistic properties of large CSPR sequence sets and uniformly random sequence sets. This fundamental study is lacking in the literature.
- We evaluate the jamming resilience of our proposal using a model implemented in MATLAB. We expose our system to an insider smart jammer and validate that the attack is mitigated. Insider AJ resilience and cross-correlation of sequences are analytically linked.

The rest of this paper is organized as follows. Section 2 gives some background on the technologies needed to build and evaluate our proposal. Section 3 defines our AJ proposal. Section 4 studies the cross-correlation properties of CSPR sequence sets. Section 5 evaluates our proposal against jamming attacks using MATLAB. Section 6 presents related work. Section 7 offers some discussion and future work perspectives. Finally, Section 8 briefly concludes this work.

2. Background

In this section, we describe three subjects relevant to this work: Cross-Correlation of Sequences, Pseudo-Random Sequence Sets, and Cryptographically Secure Pseudo-Random Number Generators.

2.1. Cross-Correlation of Sequences

2.1.1. Definitions

The correlation is a measure of the linear similarity between two sequences. If both sequences are identical, the term auto-correlation is used. Otherwise, the term Cross-Correlation (CC) is used.

The discrete *circular* CC is relevant for periodic sequences of period L. The circular CC between two periodic sequences s_1 and s_2, written as $s_1 \circledast s_2$, is defined as:

$$(s_1 \circledast s_2)[n] \stackrel{\text{def}}{=} \sum_{m=0}^{L-1} \overline{s_1[m]} s_2[m+n] \tag{1}$$

Where $\overline{s_1[m]}$ is the complex conjugate of $s_1[m]$ and n the displacement.

Normalized values of the circular CC are obtained if the result is divided by the maximum auto-correlation value. In this work, we study sequences of length L, that are used in a DSSS system as periodic sequences of period L, therefore the circular CC concept is extensively used.

2.1.2. Fast Cross-Correlation calculation

The cross-correlation of $s_1[n]$ and $s_2[n]$, written $s_1[n] \star s_2[n]$, is equivalent to the convolution of $\overline{s_1[-n]}$ and $s_2[n]$, where $\overline{s}$ corresponds to the complex conjugate of s. This equality allows to use the *convolution theorem* to obtain:

$$(s_1 \star s_2) = \mathcal{F}^{-1}\left\{\overline{\mathcal{F}\left\{s_1\right\}} \cdot \mathcal{F}\left\{s_2\right\}\right\} \tag{2}$$

Where $\cdot$ denotes point-wise multiplication, $\mathcal{F}$ stands for the Fourier transform, and $\mathcal{F}^{-1}$ for the inverse Fourier transform. This equivalence in conjunction with the fast Fourier transform allows for efficient computation of CC values in current hardware.

2.1.3. *Practical importance for Wireless Communication Systems (WCS)*

Correlation properties of the sequences have a direct impact on many fundamental properties and performance metrics of WCS. For example, low CC is desirable for Code-Division-Multiple-Access (CDMA) systems, and low auto-correlation is desirable for signal acquisition and multi-path interference rejection. Thus, the correlation of sequence sets for WCS is a widely studied subject [19, 20]. In general, families of sequence sets for WCS are designed with low correlation properties [21, 22]. One well-known example is orthogonal sequence sets: the sequences in the set have a CC of zero.

The literature on CC of sequence sets for wireless communication [23, 24, 20, 25] characterizes a given family of sequence sets by the maximum CC value of all the pairs of sequences within a set ϕ_{max}. In most cases, due to the impossibility of computing exact-values, lower bounds for ϕ_{max} are given. Some well-known bounds are Sidelnikov's [26], and Welch's [27]. For a given pair of sequences (x, y), a useful CC concept is the *cross-correlation spectra* $\phi_{x,y}$. The $\phi_{x,y}$ measures the CC values evaluated for every possible shift of one of the sequences. The set size is also a fundamental characteristic of a family of sequence sets besides the CC. Ideally, for multi-user WCS we want low ϕ_{max} and large sets. Nevertheless, there is generally a trade-off ϕ_{max} vs. set size: a low ϕ_{max} value implies a small set size.

2.2. Pseudo-Random Sequence Sets for Wireless Communication Systems

Pseudo-Random (PR) sequence sets for Wireless Communication Systems (WCS) is a well-studied topic in the literature [20, 24]. A PR sequence complies with some randomness criteria. Golomb's Randomness postulates [28] are widely accepted criteria in the WCS literature.

Several families of PR sequence sets exist. Feedback Shift Register (FSR)-based is the most prominent family of PR sequence sets. An FSR is a hardware component that consists of a chain of flip-flops sharing the same clock. The output of one flip-flop is also connected to the input of the next one. A Linear FSR (LFSR) is an FSR in which the input to the first flip-flop is a linear function of the previous FSR state. For a Non-Linear FSR (NLFSR), the input is a non-linear function of the previous FSR state.

An LFSR uses simple hardware components and produces uniformly distributed sequences with high-throughput. As a result, LFSR-families of sequence sets are historically the most used and studied [28]. However, an LFSR has weaknesses in terms of cryptanalysis due to its linear nature. For example, remaining portions of a sequence can be predicted with partial knowledge of its elements using the Berlekamp-Massey algorithm [29]. Notably, this predictability is not a desirable property from an AJ perspective. If a jammer knows the sequence of a transceiver, it can jam it in a power-efficient way [30].

Other families of PR sequence sets for WCS have gained attention in recent years due to the predictability of LFSR-based sequences. These families include Legendre/Jacobi sequences [31, 32], NLFSR-based De Bruijn sequences [25, 33, 34], and chaotic sequences [35, 36, 37]. However, these families of sequence sets for WCS still have factors that impact on their randomness properties. Either because of non-randomness-related design objectives or because of inherent functional limitations. Jacobi sequences aim at low auto-correlation properties by design. De Bruijn sequences in a set are not independent of each other, and the sets are generated with low cross-correlation design objectives. In other words, non-negligible information can be known of other De Bruijn sequences in the set if one sequence is known. Chaos theory-based sequences have practical-use design challenges [37]. For example, chaotic sequences are inherently non-periodic, and this forces either robust-synchronization of the chaotic system state, or complex non-coherent methods for demodulation. Besides, their use for cryptography use is proven to be immature and broken [38, 39], which implies their randomness properties are compromised.

2.3. Cryptographically Secure Pseudo-Random Number Generators and Stream Ciphers

A Cryptographically Secure Pseudo-Random Number Generator (CSPRNG) is a functional block that takes some input parameters (i.e., a *key* and a *nonce*) and produces pseudo-random output suitable for cryptographic use [40, 41]. The same input parameters will consistently produce the same output.

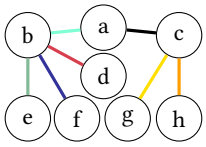
Stream ciphers [42] are symmetric ciphers. With a given *key*, a stream cipher generates a cryptographically secure pseudo-random stream of bits called a *keystream*. The keystream is independent of the message to be encrypted. The ChaCha20 [43] is a stream cipher designed to be fast on pure-software implementations. It is used as a state-of-the-art cipher in Internet security protocols such as IKE-IPsec [44] and Transport Layer Security [45]. Cha-Cha20 uses a 256-bit *key* and a 64-bit *nonce* as inputs. It can output 2^{41} bits (274 GBytes) of pseudo-random data (keystream). In this work, we use Cha-Cha20 as a CSPRNG. First, because in terms of security, it is a well-established stream cipher. Second, because it is software-optimized [46], and this offers great flexibility for the dynamic IoT systems we target. For example, on already deployed IoT nodes we can replace it with another state-of-the-art cipher in the future.

3. Proposal

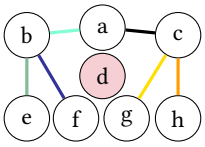
In this section, we present an MTD mechanism targeted at the *physical layer* (PHY) of a communication system. The main objective of our proposal is to improve the jamming resilience of IoT networks. In particular, we want to proactively mitigate the impact on the system of insider-node jamming attacks.

3.1. Overview

An example of a target IoT network is illustrated in Fig. 1a: circles represent IoT nodes, and edges are communication links. Every link between a pair of communicating no-



(a) IoT AJ MTD network: every pair of nodes use unique PHY parameters.



(b) Insider attack: node d compromised. Jamming resilient PHY-links.

Figure 1: IoT MTD Network

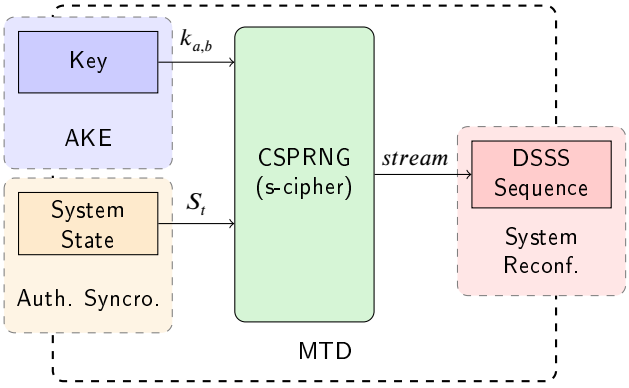


Figure 2: MTD Proposal: Cryptographically Secure PR-DSSS.

des in the IoT network has distinct physical layer parameters defined by our MTD proposal. These pairwise parameters are only known by each pair. This physical link diversity is represented with different edge colors in Fig. 1a.

To illustrate the potential AJ advantages of such a system, consider an *insider attack*. In this kind of attack, one of the legitimate nodes in a network becomes an adversary. Fig.1b shows node d, as an insider attacker in an IoT network. In a legacy IoT network, all the nodes share the physical layer parameters are shared by ; therefore, an insider attacker can potentially become a very power-efficient jammer. In our proposed MTD system, an insider attacker's jamming impact is mitigated because it has no perfect knowledge of the physical layer parameters for every link in the network. As stated before, *every link* between a pair of nodes has unique physical layer parameters known only by each pair.

3.2. MTD System: CSPR Sequences for DSSS

The IoT network has Direct-Sequence Spread-Spectrum (DSSS) capabilities. We define our system using three elements helpful to design an MTD technique [7]. (1) WHAT to move: define the moving parameter. (2) HOW to move: define the procedure to change the parameter. (3) WHEN to move: define the condition of change.

WHAT. The moving parameter of our proposal is the DSSS spreading sequence ss_{mtd}.

HOW. We assume two pre-requisites between any pair of communicating nodes, a and b:

1. They share a cryptographic Key $k_{a,b}$
2. They share an authenticated synchronized System State S_t

A given pair of IoT nodes will have a unique cryptographic Key $k_{a,b}$, whereas all the nodes in the network can share the System State S_t. S_t is a cryptographic *nonce* (a number used once) with no randomness requisites.

The core of our MTD proposal depends on a Cryptographically Secure Pseudo-Random Number Generator (CSPRNG). We propose to use stream-ciphers [42] as CSPRNG, particularly ChaCha20 [43].

We describe the procedure for a given pair of nodes a and b. First, we use the Key $k_{a,b}$, and the System State S_t as inputs for the CSPRNG. Second, we apply a simple transformation to the CSPRNG output: truncate to L bits. Finally, the result is used as a periodic spreading sequence ss_{mtd} for DSSS modulation between the pair of nodes. Fig. 2 illustrates our proposed MTD mechanism from a single IoT node perspective.

WHEN. The *movement* concept of the MTD system is determined by the System State S_t variable. For a given instant in time t, all IoT nodes should agree with the value of S_t. The frequency of change f_S of S_t will determine the frequency of movement of the MTD system. If S_t changes, the nodes must recalculate the DSSS sequence. Unlike $k_{a,b}$, the value S_t does not need to be secret, only authenticated. One natural candidate for S_t is the Coordinated Universal Time (UTC). The authenticated time synchronization can be coarse-grained in most cases because the traffic in constrained IoT networks is sporadic (e.g., one packet transmission per minute, hour, or even day).

3.3. Implications of PHY Randomization

The Cryptographically Secure Pseudo-Randomization of DSSS spreading-sequences effectively mitigates insider jamming attacks, as will be evaluated in Sec. 5. This insider AJ resilience comes with a trade-off in terms of multi-user performance. The cross-correlation values of the spreading-sequences in a DSSS system determine the multi-user performance, the lower, the better. Because our proposal randomizes the sequences in a decentralized and independent way, there are no low-cross-correlation guarantees for the system.

However, we can statistically study the cross-correlation values of large CSPR sequence sets. Furthermore, because of the good randomness properties of CSPRNG, any given CSPR sequence set can be probabilistically characterized. The cross-correlation of the CSPR sequences is not only the determining factor of multi-user performance but also of the insider AJ mitigation. This cross-correlation statistical study is presented in the following Section 4.

4. Cross-Correlation of CSPR Sequence Sets

In this section, we study the statistical distribution of Cross-Correlation (CC) values of large Cryptographically Secure Pseudo-Random (CSPR) sequence sets. We prioritize an empirical approach. First, we generate large CSPR

sequence sets. Second, we calculate the CC of all pairwise sequence combinations in the set. Finally, we calculate the Empirical Cumulative Distribution Function (ECDF) of the CC values. We also provide analytical results that validate the empirical study. We conclude this section with a CC comparative study with other families of PR sequence sets.

4.1. Motivation

For a given family of sequence sets for Wireless Communication Systems, the Cross-Correlation and the Cardinality (i.e., number of elements) of the sets are two of the most important characteristics. They determine the multi-user capabilities of the system. As stated in Sec. 3.3, only probabilistic statements can be made about the cross-correlation values of CSPR sequence sets. To the best of our knowledge, no work in the literature studies this problem with enough depth. This section deals with this fundamental study. These results are used in Sec. 5 to characterize the AJ capabilities of our proposal analytically.

4.2. Sequence Sets Generation

Let $S_{(L,CSPRNG)}$ be a generated sequence set. The sequences in the set are binary sequences of a fixed length L, and were generated using the same CSPRNG. The characteristics and generation-input parameters of the sets are the following:

- Cardinality (set size): 1024
- L (bits): {128, 256, 512, 768, 1024, 2048, 4096, 8192, 16384, 32768}
- CSPRNG: {ChaCha20 [43], AES-CTR[1]}
- CSPRNG Inputs:
  - Key: $\{(0)_{10}, (1)_{10}, (2)_{10}, ..., (1023)_{10}\}$ (256-bit length and zero-padded)
  - System State: {0}

For example, the set $S_{(128,ChaCha20)}$ is composed of 1024 sequences $\{s_0, s_1, ..., s_{1023}\}$ of length 128. A sequence s_i ($i \in \{0, 1, ..., 1023\}$) is generated using ChaCha20 as CSPRNG. And its inputs are: $Key = (i)_{10}$, and $System\ State = 0$; the output is truncated to *128* bits.

A generated binary sequence $\{b_1, ..., b_L\}$ has unipolar encoding with elements $b_i \in \{1, 0\}$, from now on we will work with sequences in bipolar encoding where $b_i \in \{1, -1\}$. Also, a given sequence of length L will be used in our communication system as a periodic sequence with period L; thus, length and period are equivalent terms in the rest of the section.

4.3. Normalized Cross-Correlation Calculation

The circular Cross-Correlation (CC) (Eq. 1) between two bipolar binary sequences of length L, evaluated at a displacement n, takes integer values comprised in $\{-L, ..., 0, ..., L\}$. In order to compare CC properties of sequences of different length L, the Normalized CC is useful. It is obtained dividing the CC value by the maximum possible value, in our case L. Furthermore, we take the absolute value of this result, as in terms of sequence-signal interference, the sign of the CC value is irrelevant. The expected Normalized CC values will be comprised in $\{0, \frac{1}{L}, \frac{2}{L}, ..., 1\}$, where the value of 0 is associated with an orthogonal sequence, and 1 with the max value, i.e., the same sequence.

Let $|NCC_{(x,y)[n]}|$ be the absolute value of the Normalized circular CC. NCC for short. For a given pair of sequences (x, y) of length L, and evaluated at a fixed displacement (time shift) $n \in \{0, 1, ..., L-1\}$, this is defined as:

$$|NCC_{(x,y)[n]}| = \left| \frac{1}{L} \sum_{m=0}^{L-1} \overline{x[m]} y[m+n] \right| \tag{3}$$

NCC is a scalar value. For every generated set $S_{(L,\text{CSPRNG})}$, we calculate NCC for every pair of sequences (x, y) in the set, and for every displacement $n \in \{0, ..., L-1\}$. Let $CC\_S_{(L,CSPRNG)}$ be the set that contains all the calculated NCC. This new set contains at least 50 million NCC elements.

To calculate the NCC values (Eq. 3), we use the convolution theorem (Eq. 2) in conjunction with fast Fourier transforms using an Intel Core i7-6600U CPU @ 2.60GHz x 4 with 16 GB RAM. For sets of large sequence length L we lowered the cardinality, but this does not affect the statistical relevance of the results[2].

4.4. Statistical Results and Probability Analysis

In Fig. 3 we show the Empirical Cumulative Distribution Functions (ECDFs) of NCC values of ChaCha20 generated sets $S_{L,ChaCha20}$. Every ECDF was calculated with between 50 and 500 million NCC values. The ECDFs of AES-CTR generated sets are visually indistinguishable from the ChaCha20. The statistical similarity is expected, as by design a CSPRNG is indistinguishable from a *True RNG*[3]; and, by the Glivenko-Cantelli Theorem [47], both ECDFs converge to the same Cumulative Distribution Function (CDF).

It can be observed in Fig. 3, that the greater the length L of the sequences in the set S_L, the lower the NCC values for almost all the percentiles (i.e., x-value for a given cumulative probability) of the ECDFs. This was expected, informally the longer the pseudo-random sequences, the lower the probability of two sequences being similar to each other. For example, the 80th percentile $P_{80} = k$ (i.e., 80% of the NCC values are $\leq k$) is for S_{128}, $k \approx 0.12$; for S_{256}, $k \approx 0.08$; and for S_{512}, $k \approx 0.06$. This is not true for every percentile. For some percentiles lesser than P_{15}, we can see that the ECDFs intersect each other. The step-nature of the ECDFs explains these counter-intuitive results. It is worth noting that this step-nature of the ECDFs is not due to a limited number of

[1] AES is a block cipher, but in CTR mode behaves as a stream cipher.

[2] Glivenko-Cantelli's Theorem [47] states that the ECDF of a random variable converges uniformly to the CDF of the underlying-unknown distribution.

[3] A generator of truly uniformly distributed bit string -*Bernoulli process* with $p = 0.5$-, or in cryptographic terms a *random oracle*.

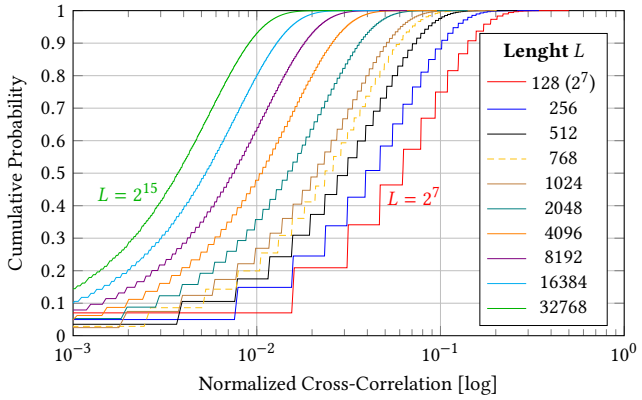


Figure 3: ECDF of Normalized CC of ChaCha20-generated sequence sets, for different sequence length L.

sample variables NCC (at least 50 million), but to the inherent discrete values the NCC takes for binary sequences of a fixed length L.

Ultimately, we want to predict the NCC distribution of any given CSPR sequence set. The randomness properties of CSPRNGs and the Glivenko-Cantelli Theorem give us strong statistical guarantees that any random set of CSPR sequences will follow the ECDFs shown in Fig. 3. Furthermore, we strengthen this statement with a pure analytical-probability approach that is found in the next Section 4.5. The single hypothesis is that a CSPRNG output resembles a true uniformly random process where every produced bit has an equal probability of being 1 or -1. The results validate the statistical analysis. For $L = 128$ the ChaCha20 ECDF corresponds to the Analytical CDF with a point-wise precision of ± 0.0001. An important analytical result is that a given percentile P_n is a function of $\sqrt{L}$. This result will have practical AJ system design consequences, as will be explored in Sec. 5.

4.5. Analytical Cross-Correlation Distribution of CSPR Sequences

A CSPRNG of length L must be statistically indistinguishable from a Bernoulli Process of L trials. Using this equivalence, we develop an analytical probabilistic study of the CC of CSPR sequences.

Let $s = \{b_1, b_2, ..., b_L\}$ be a CSPR binary sequence of length L, where every bit b_i is a random variable (r.v.) that follows a Bernoulli distribution with $p = 0.5$.

Let $s_1 = \{x_1, ..., x_L\}$ and $s_2 = \{y_1, ..., y_L\}$ be two independent CSPR binary sequences, with L even. The circular cross-correlation of s_1 and s_2, evaluated at $n = 0$, is:

$$(s_1 \circledast s_2)[0] \overset{\text{def}}{=} \sum_{m=1}^{L} \overline{s_1[m]} s_2[m+0] = \sum_{m=1}^{L} x_m y_m = \sum_{m=1}^{L} b_m \quad (4)$$

Where b_m is a r.v. that also follows a Bernoulli distribution[4] with p=0.5. The probability distribution of the sum of two or more independent r.v. is equivalent to the convolution of their individual distributions. Particularly, the sum of two Bernoulli r.v. results in a r.v. with Binomial Distribution of 2 trials. It is well known that $\sum_{n=1}^{L} Bernoulli(p) \sim B(n, p)$, where $B(n, p)$ is a Binomial where n is the number of trials. With this result, we develop Eq. 4. The resulting random variable follows a Binomial distribution $B(L, 0.5)$. However, this is true only if the original domain-support of the Bernoulli distribution where $k \in \{0, 1\}$, but in our signal processing setting, we use binary bipolar values $k \in \{-1, 1\}$. We apply a change of variable (c.v.) to transform the result of the studied r.v. $X \sim B(L, 0.5)$ with support $x \in \{0, 1, 2, 3, \dots, L\}$, to our signal processing setting support where the cross-correlation result will take only even values $y \in \{-L, \dots, -4, -2, 0, 2, 4, \dots, L\}$. The c.v. is $y = 2(x - \frac{L}{2}) \longrightarrow x = \frac{y+L}{2}$. The resulting r.v. represents the CC value of two CSPR sequences, and its Probability Mass Function (PMF) is symmetrical with respect to zero. We need one more transformation because we are interested in the absolute value of the cross-correlation $|CC|$. The $|CC|$ support will be $k \in \{0, 2, 4, \dots, L\}$ with k even. Because of the symmetry of the CC r.v. PMF , the $|CC|$ r.v. PMF is straightforward; we double the probability of all the positive values, except for zero. More precisely, the Probability Mass Function of the absolute cross-correlation $|CC|$ of two CSPR sequences of even length L is, as a function of the taken value $k \in \mathbb{N}_0$:

$$\mathbb{P}(|CC| = k) = \begin{cases} \dfrac{L!}{(L/2)!(L/2)!} \left(\dfrac{1}{2}\right)^L & \text{if } k = 0, \\ \dfrac{L!}{\left(\frac{L+k}{2}\right)! \left(\frac{L-k}{2}\right)!} \left(\dfrac{1}{2}\right)^L \times 2 & \text{if } 0 < k \leq L, \, k \text{ even}, \\ 0 & \text{otherwise.} \end{cases}$$

The Cumulative Distribution Function (CDF) can be either calculated directly or expressed in terms of the regularized incomplete beta function. However, we take another approach to further describe the probabilistic properties of the $|CC|$ of CSPR sequences.

Using the De Moivre-Laplace theorem, we can approximate the Binomial distribution $X \sim B(L, 0.5)$ with a Normal distribution $N(\mu, \sigma)$ of mean $\mu = L/2$, and standard deviation $\sigma = \sqrt{L/4}$. This results in $X \sim B(L, 0.5) \sim N(\frac{L}{2}, \sqrt{L/4})$. We apply the same c.v. as in the discrete case to transform the result to our CC r.v. domain, a horizontal shift of $-L/2$, and a horizontal dilation by a factor of 2. This results in $CC \sim N(0, \sqrt{L})$. This continuous approximation takes into account the odd values of the horizontal axis. Because our domain support $k \in \{0, 2, 4, \dots, L\}$ only has even values, we need to multiply $\times 2$ the approximated probability, excepting for $k = 0$. A last transformation is needed to obtain the absolute value and represent the $|CC|$ r.v., the result closely resembles a half-normal distribution. Finally, the approximation of the Probability Mass Function of $|CC|$ is:

[4] Multiplication of two independent Bernoulli variables is also a Bernoulli.

Support	$k \in \{0, 2, 4, \ldots, L\}$
Mean	$\approx \sqrt{L}\sqrt{2/\pi}$
Variance	$\approx L(1 - \sqrt{2/\pi})$
Median	$= \lfloor\sqrt{L/2}\rfloor$or$\lceil\sqrt{L/2}\rceil$ (the even value)
Mode	$= 2$
CDF $F(k)$	$\approx \text{erf}\left(\frac{k}{\sqrt{L}\sqrt{2}}\right)$

Table 1
Properties of the $|CC|$ of CSPR seq. of length L

PR Family	NCC_{max}	Set Size	Seq. Length L
Gold	0.130	257	255
Kasami (large set)	0.130	4 112	255
Kasami (small set)	0.067	16	255
De Bruijn (low-CC [25])	0.130	16	256
CSPR-ChaCha20 (C.I. 95%)	0.305	16	256

Table 2
$\mathbf{NCC_{max}}$ for different families of PR seq. sets

$$\mathbb{P}(|CC| = k) \approx \begin{cases} \frac{\sqrt{2/\pi}}{\sqrt{L}} & \text{if } k = 0, \\ \frac{\sqrt{2/\pi}}{\sqrt{L}} e^{-\frac{k^2}{2L}} \times 2 & \text{if } 0 < k \leq L, \text{ and } k \text{ even}, \\ 0 & \text{otherwise.} \end{cases}$$

We validated using Wolfram Mathematica that the approximation is correct at least with 3 significant digits for $L \geq 128$. Some important characteristics of the $|CC|$ distribution of CSPR sequences of length L are shown in Table 1.

Finally, for having results that correspond to the Normalized $|CC|$ (NCC), with support $k' \in \{0, \frac{2}{L}, \frac{4}{L}, .., 1\}$, the change of variable $k = k'L$ should be done.

4.6. Comparison with NCC of other PR families

In this section, we compare the obtained results against other families of Pseudo-Random sequence sets. The literature generally characterizes a given family with the NCC_{max} of all the sequences in a set.

The NCC_{max} expresses the worse-case value of the NCC of all pairwise sequence combinations in a set, for every relative sequence displacement. All other pairwise combinations have lower NCC values. When realized, the NCC_{max} affects two pairs of communicating nodes, not the whole system. An NCC_{max} will be realized when the specific pair(s) of nodes communicate at the same time, and for a specific relative sequence time-shift (displacement).

For PR families in the bibliography, there is hard-bounds by design for the NCC_{max}. For CSPR sequence sets, we have no hard-bounds (i.e., any value is possible, albeit with different probability), but a probabilistic estimation can be given. In Table 2, we show NCC_{max} values of CSPR[5] sequence sets compared with other PR families. For CSPR-ChaCha20 sequences of length $L = 256$, the NCC mean value is ≈ 0.050[6] and the median equals 0.044194174.

The NCC_{max} for CSPR sequences with a Confidence Interval (C.I.) 95% is more than double compared with other families. This higher NCC is undesirable for constant high-throughput systems with a centralized-star topology (i.e., cellular networks, where a central node communicates with all others). For the IoT use-case, where traffic is packet-based and sporadic, the impact of the theoretical NCC_{max} on the system performance is not that relevant because: (1) the probability of realization of the event is low (0.00088796%[7]), and (2) if it happens, the impact on the system performance will be over a single packet. Thus, arguably, the statistical distribution of cross-correlation of sequences is a better suited tool than the single-value NCC_{max} (i.e., a low-probability and short-lived event) to predict the expected (e.g., mean, average) performance of packet-based low-throughput systems like the IoT.

5. Evaluation: AJ Resilience of Proposal

In this section, we present numerical results that measure the AJ resilience of our proposal. System AJ resilience is measured in terms of Bit-Error-Rate (BER) as a function of jammer signal power. We use MATLAB to simulate a multi-node DSSS system. First, we present the system and attacker model. Then, we evaluate the system AJ resilience against two types of attackers (jammers). (A) A Broadband Noise Jammer that represents a baseline for jammer power efficiency, i.e., any other DSSS jamming strategy will be preferable for the attacker. (B) An *Insider Smart Jammer* that represents an upper-bound for jammer power efficiency. The latter jamming scenario is the most relevant. It instantiates our insider-node jamming attack hypothesis and measures the degree of jamming mitigation of our proposed CSPRNG-based DSSS system. The Insider Smart Jammer AJ resilience is analytically linked to the cross-correlation properties studied in Section 4.

5.1. System Model

The system model is shown in Fig. 4. The system is composed of n transmitter nodes (TX-nodes), a jammer, one receiver node (RX-node), and a channel modeled as Additive White Gaussian Noise (AWGN). The Binary Phase-Shift Keying (BPSK) modulation in our setting is the conversion from unipolar {1,0} to bipolar {1,-1} of a signal sample. The communication process is as follows. (1) A TX-node sends a random digital signal modulated with BPSK. (2) A binary Spreading-Sequence (SS) is applied after BPSK, and

[5] $P_{99.999926}(NCC) = 0.305$, combined probability for 256 values of the spectra -taken as, *i.i.d* variables- and for a set of 16 sequences $(0.99999926)^{256\times\binom{16}{2}} = 0.9555$.

[6] Standard deviation = 0.449

[7] For a given instant in time, with the 16 sequences being used at the same time, $P(NCC_{\text{at least 1 pair}} \geq NCC_{max}) \leq 1 - (0.99999926)^{120}$

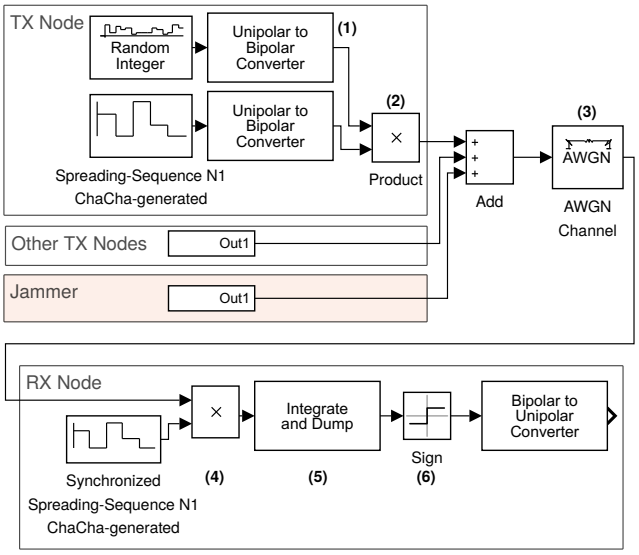


Figure 4: DSSS-CDMA System Model

the transmission is in base-band. (3) Other signals and AWGN are added. The RX-node demodulates the received signal by (4) de-spreading it using a synchronized version of the SS, (5) applying an *integrate-and-dump* correlator, and (6) making a decision based on the sign of the signal to determine the BPSK symbol.

About the AWGN channel model: The channel model does not account for fading, frequency selectivity, multi-path, nonlinearity, or dispersion. This simplification is not in demerit of the jammer. An AWGN channel highlights the relationship between the jammer and the nodes' signal power in the demodulation process. Most power-independent phenomena on a more realistic channel will be to the disadvantage of the jammer power efficiency because a receiver will be optimized to compensate the channel's effect on signals from legit nodes. For example, a frequency and phase-shifted jamming signal will have less impact on the induced BER at the demodulation process of a legit node, as compared to an in-phase and frequency version of the same jamming signal.

5.2. Attacker Model

The attacker is a *jammer*. The jammer has access to the communication medium and can insert arbitrary signals. Its goal is to produce errors in the demodulation process of the receiver. Eavesdropping, tampering, or forgery of information is not a goal. The jammer has enough energy, power, and bandwidth to cover the entire band of the system with a signal of arbitrary power. The jammer can be further defined by the type of signal it inserts in the channel. We evaluate two types of jammers: a Broadband Noise Jammer and an Insider Smart Jammer (coherent and synchronous), both further detailed in Sections 5.3 and 5.4, respectively.

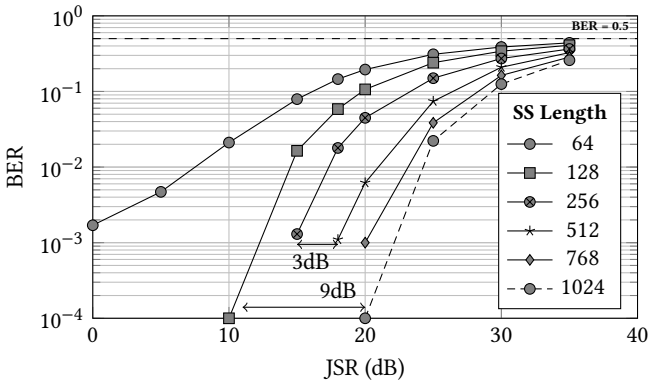


Figure 5: BBN jammer resilience

5.3. Baseline Evaluation: Broadband Noise Jammer

A Broadband Noise (BBN) jammer places a random noise signal over the full width of the TX-node communication spectrum. A BBN strategy raises the noise level at the receiver and is a direct attack on the channel capacity of any communication system [48]. A BBN represents a *baseline* in terms of jammer power efficiency for an *effective* jamming strategy against a DSSS system[8][48].

Attacker Hypothesis The jammer knows the center frequency and bandwidth of the signal.

Simulation We used MATLAB/Simulink R2017a to model the system described in Sec. 5.1. There are 4 TX-nodes. We chose that number because it is representative of a one-hop-link in an IoT mesh network. Even if an IoT network has more nodes, for a given node, only nodes physically close have a signal-power relevance. Every TX-node constantly sends random binary data sampled at 1000 times/sec (1kbps), and we simulate 10 seconds (i.e., 10000 bits). Every TX-node uses a different Spreading-Sequence (SS). The jammer power J_p is measured in relative terms against a single TX-node power S_p, this ratio is expressed as the Jamming-to-Signal Ratio (JSR) $= J_p/S_p$ in dB[9]. The AWGN channel power is included in the JSR. We calculate the BER at the RX-node. We repeat the simulation with different SS lengths. The results are shown in Fig. 5, every data-point is a simulation run.

Analysis Obtained results are consistent with the DSSS AJ theory [48]. When we double the SS length, we also double the signal bandwidth, and we obtain a ≈+3dB gain in resilience to Gaussian Noise for a fixed BER. The exception is the SS codes of length 64 for JSR<15dB. We explain this weaker resilience as a result of the higher Cross-Correlation values of the SS of the other legit TX-nodes. For this work, we consider a Bit-Error-Rate (BER) ≤ 0.1 to be acceptable

[8] Other jamming strategies such as Narrowband noise are not considered *effective*.

[9] Power Decibel: $10\log_{10}(P1/P2)$

for a digital communication system under jamming [48].

5.4. Upper-Bound Evaluation: Insider Smart Jammer

In this section, we define and model an *insider smart jammer*. We find it useful to explicit again our setting and evaluation goal.

Setting and Goals The jammer is not limited in terms of energy; i.e., it can apply a jamming signal with power JSR persistently in time. There is no reactive AJ mechanism in our current evaluation. Thus, it is inevitable that for a given JSR, the jammer will *defeat* ($BER \geq 0.1$) the system. **An insider smart jammer will defeat most AJ systems with a JSR≈0**. When the insider knows the AJ parameters of a single node, he can compromise the whole system's parameters. These AJ systems will be compromised either because the nodes share the same AJ parameters (e.g., hopping-schedule), or because the system uses a PR sequence set with no cryptographically secure pseudo-randomness properties (e.g., LFSR-generated, low-CC De Bruijn). In our proposal, the knowledge of one spreading sequence by an attacker does not imply the compromise of the other spreading sequences in the system. From a system perspective then, it is relevant to study the case in which the attacker has gained knowledge of the spreading sequence of *one node*. Then, measure the impact over the BER of the rest of the nodes in the system. Our system proposal was designed with the main objective of proactively mitigating this insider-node attack, and this section measures to which degree this is achieved.

The insider smart jammer represents an upper-bound for jammer power efficiency, under certain hypothesis.

Attacker Hypothesis The jammer:

- Has perfect knowledge of the system (e.g., BPSK), except for the spreading sequences (SS) used by the nodes.
- Is synchronized (time) and in-phase (*coherent*) with the SS of the system.
- Knows the SS of *one* node, i.e., a compromised node.
- Can not compromise the SS of other nodes[10].

Simulation The set-up is the same as the BBN jammer. The jammer is modeled as a TX-node, which uses the SS of one compromised node. We measure the BER of non-compromised nodes for different values of JSR. The AWGN channel has a Signal-to-Noise-Ratio (SNR) > 0 compared with a TX-node and is negligible. We plot the BBN jammer results for reference. The results are shown in Fig. 6.

Analysis In all cases, the BER vs. JSR curve for individual nodes shows similar behavior. Let J_n be a threshold value unique to a $node_n$. Then, $BER = 0$ if $JSR < J_n$; followed by a steep positive slope at $JSR \approx J_n$, from $BER = 0$ to 0.5

[10]See Appendix. A for justification of this hypothesis.

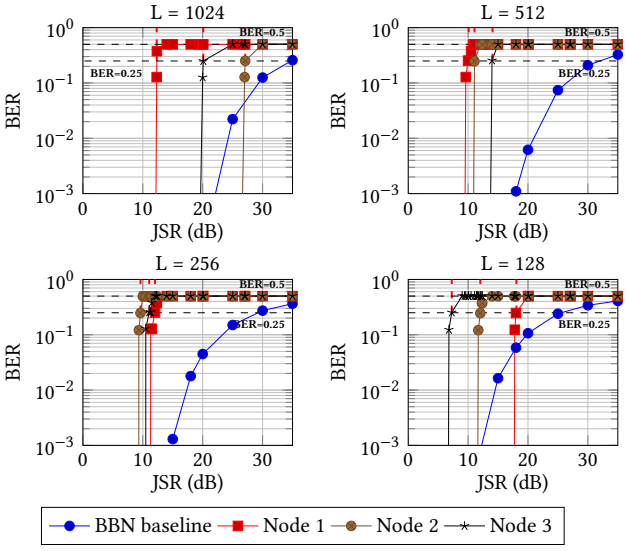


Figure 6: Smart jammer resilience for different SS Lengths L.

; finally, a constant $BER = 0.5$ for $JSR > J_n + \Delta JSR$. Not surprisingly, this threshold value J_n is related to the Cross-Correlation (CC) value between the spreading sequences of the jammer and the $node_n$.

In the following, we provide an informal explanation of the relationship between CC and $JSR_{BER=0.25}$ and refer to [49] for a general analytical expression. This relationship is valid for an integrate-and-dump DSSS correlator, and BPSK modulation. In our case, the jammer is in phase and frequency with the target signal. This maximizes the impact of the jammer in the target signal. Suppose that (A) the Normalized CC (NCC) of the spreading-sequences is 1 (i.e., is the same sequence). In that case, a jammer with equal power as the TX-node ($JSR = 0dB$) sending random bits will achieve a $BER = 0.25$ at the RX-node. This BER value is explained because of the RX-node with equal probability $= 0.5$ decoding either the TX-node's bit or the jammers' bit, which in turn has a probability $= 0.5$ of being the same as the TX-node bit. This gives a total probability of decoding the correct bit of $p = 0.75 \rightarrow BER = 0.25$. (B) In the other extreme case, if the NCC is 0 (i.e., orthogonal sequences) theoretically there is no value for JSR that will affect the BER. (C) In the most general case, for an NCC between 0 and 1, say an NCC of a ratio $\frac{1}{N}$, the jammer needs N times more power to achieve the same effect as a $NCC = 1$. We formalize this relationship that derives from the work of [49], in the following Eq. 5:

$$JSR_{BER=0.25}(NCC) = 10\log_{10}(\frac{1}{NCC})[dB] \quad (5)$$

For example, if the $NCC = 0.1$ between the sequences of a jammer and a node, a jammer needs 10 times more signal power than the TX-node ($JSR = 10dB$) to achieve a $BER = 0.25$ at the RX-node.

For each node in our evaluation, we calculated the NCC and the theoretical JSR for $BER = 0.25$. In Fig. 6 we mark this theoretical value with a vertical dashed line from

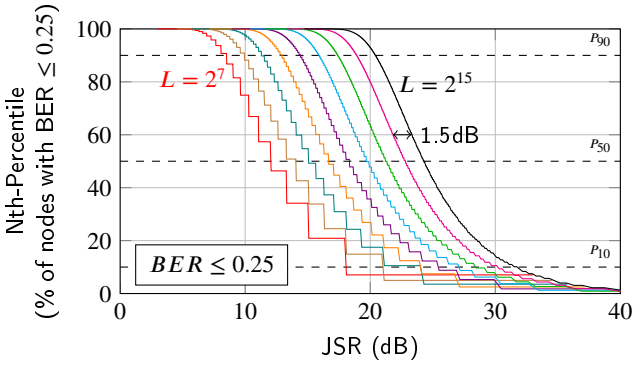


Figure 7: Percentiles for nodes with predicted $BER \leq 0.25$ as a function of an insider smart jammer power JSR, for different spreading sequences (SS) lengths $L \in \{2^7, 2^8, ..., 2^{15}\}$.

$BER = 0.25$ to 1. In all cases, the simulated results correspond to the predicted theoretical values.

Synthesis We want to characterize the insider smart jammer resilience of a generic system implementing our proposal. In order to do so, we use (1) the study of the NCC for CSPR sequence sets from Section 4, and (2) Equation 5 that analytically relates $NCC \longleftrightarrow JSR_{BER=0.25}$. With these two elements, we can probabilistically describe the smart jamming resilience of BPSK-DSSS systems that use CSPR sequence sets. We transform the empirical values of NCC of CSPR sequences using Equation 5. The obtained results are shown in Fig. 7. With this ECDF representation, we can quickly determine the percentage (i.e., percentiles P_i) of the nodes in a generic system that will have a $BER \leq 0.25$, for a given JSR and sequence length L. An important observation is that contiguous curves in Fig. 7, where L is related by a factor of 2, are approximately +1.5dB apart ($\sqrt{2}$ in linear terms). In Appendix B, we confirm this observation empirically, and the results of Section 4.5 validate this relationship analytically.

Approximation of behavior for $BER \leq 0.1$ As stated before, an AJ resilient system should have a $BER \leq 0.1$ under a jamming attack. However, our study is relevant for $BER \leq 0.25$ because Eq. 5 relates $NCC \longleftrightarrow JSR_{BER=0.25}$. Nodes with BER values between $(0.1, 0.25]$ should be excluded in our study. We need an expression that relates $NCC \longleftrightarrow JSR_{BER=0.10}$. We were unable to find an exact analytical expression that predicted the simulated $BER = 0.10$ as a function of NCC. Yet, we found an upper-bound approximation. If we apply $-0.5dB$ to the analytical $JSR_{BER=0.25}$, in all simulated cases $BER = 0.0000$. Therefore, results in Fig. 7 will approximate P($BER \leq 0.1$), if we apply an horizontal-displacement $\Delta = -0.5dB$ to the ECDFs. This approximation is used to draw general conclusions about insider jammer AJ resilience of our system in the next section.

5.5. Evaluation Summary

Results from this section provide useful insights about how a system implementing our proposal will perform against jamming in general. We defined two scenarios that represent a baseline and an upper bound for jammer power efficiency. Any other effective jamming strategy against a DSSS system will fall in-between these.

The length L of the Spreading-Sequences (SS) is the most important factor for jamming resilience. Longer SS provide better BER of the overall system in both scenarios. However, increasing L comes at a non-negligible cost: either bandwidth is increased for a fixed bit-rate, or bit-rate is lowered for a fixed bandwidth.

With the results from this section, we can quantify the jamming resilience we add to a CSPR-DSSS system by increasing L. With this information, a system designer can choose an appropriate trade-off between AJ and bandwidth/-bit-rate. For the BBN jammer, results are well-known from DSSS theory: if we double L, the jammer needs approximately double the power (JSR $+3dB$). Also, BBN jammer affects all the nodes of the network homogeneously independently of the SS sequences. The insider smart jammer scenario is different. The smart jammer affects each node differently. Hence, we can only give a probabilistic characterization of the AJ capabilities of the system. The cross-correlation properties of the CSPR SS are a fundamental factor that determines the system resilience, Fig. 7 resumes the link between the statistical properties of CSPR SS and smart jammer resilience.

To summarize, the provided jamming resilience of our proposal as a function of the CSPR sequences length L is:

- BBN Jammer resilience $\mathcal{O}(L)$
- Smart Jammer resilience $\mathcal{O}(\sqrt{L})$ [11]

Resilience performance against *any other* jamming attack should fall-in between those values. For the sake of completeness, for a fixed bit-rate, the bandwidth of a DSSS system is $\mathcal{O}(L)$; and for a fixed bandwidth, the bit-rate of a DSSS system is $\mathcal{O}(1/L)$.

6. Related Work

This section reviews related work in two topics. First, we review work on correlation studies of Pseudo-Random (PR) sequences. Second, we introduce some works that use Cryptographically Secure Pseudo-Random (CSPR) mechanisms for the design of Anti-Jamming (AJ) Wireless Communicating Systems (WCS).

6.1. Correlation of Pseudo-Random Sequences

The study of correlation properties of PR sequence sets has been focused on the families highlighted in Sec. 2.2. Those families are LSFR-based, De Bruijn sequences [25, 33, 34], Legendre/Jacobi sequences [31, 32], and Chaotic Sequences [35, 36, 37]. The most studied family is LSFR-based. A classical Cross-Correlation (CC) reference is the work of Swarte et al. [23]. More recent work is given by

[11] Justification for the square-root relationship between smart jamming resilience and L is given in Sec. 4.5 and Appendix B.

Zepernick et al. [24], it also covers other PR families. For a particular PR family, we refer the reader to the corresponding cited works of a given family. The CSPR Sequence Sets we used throughout this work does not correspond to any PR family in the literature.

Notwithstanding the fundamental and well-studied topic of the uniformly random probability distribution, there is a lack of studies on the CC properties of uniformly-distributed sequence sets. Schotten et al. [21] gave an analytical formula for the auto-correlation of true-random sequences (i.e., a Bernoulli process) with the assumption of Golay's ergodicity postulate. However, to the best of our knowledge, no such analytical equivalent exists for CC characterization of uniformly-random sequence sets. Pöpper et al. [50] provided an empirical characterization of the CC of *random codes*[12], giving three percentiles ($P_{95}, P_{99.99}, P_{100}$) for sequences of length [$128 \leq L \leq 1024$], and sets with cardinality 1000. From the pure-mathematics field, Kuipers et al. [51] studied properties of an operation similar to the *convolution* on topological spaces linked to the uniform distribution. However, Kuipers et al. focused on demonstrating linear relationships between the operations.

6.2. CSPR-based AJ WCS

The closest proposal in the literature to ours is NATO's unclassified work by F. Hermanns [52] (German Patent [53]). It proposed to use AES-OFB cipher[13] output as *code-hopping* (CH) sequences. The proposal is a hybrid DSSS-FHSS system. Unlike classical DSSS, where a central carrier frequency is known, CH also 'hops' the center carrier frequency, thus doing FHSS. If originally available bandwidth was unused by the transceivers, this approach effectively increases AJ resilience. However, if the transceivers already used DSSS over the whole available bandwidth, the gain of this approach is yet to be evaluated. Hermanns evaluates the linear-complexity of AES-OFB sequences, empirically measures the auto-correlation spectra for a sequence, and evaluates the multi-user and AJ performances with a simulated system (the simulation platform is not disclosed). The jammer model is not fully specified for the AJ evaluation. We assume a synchronous and coherent jammer model was used because its results were consistent with ours. For a sequence length of 1000, it measured a *security gain* of +10 dB at the $BER = 10^{-3}$ level.

T. Song et al. [54] focus on a single link of a CDMA communication under *disguised jamming* (equivalent to the smart jammer, but not an insider). They propose to use AES (mode of operation not detailed) to encrypt LFSR-generated PN Sequences. They use the term *Secure Scrambling*[14] to refer to this operation. The legit parties share secret keys of 128, 192, or 256 bits. They focus on an analytic study of the impact of a disguised jammer who does not know the sequence, and the system using the Arbitrarily Varying Channel (AVC) model. It is relevant to note, in the context of our work, that the generic analytical results obtained by Song et al. in the AVC model can also be applied to a single-link of our proposal, as an AES-output should behave as a CSPR output. They conclude that the secure scrambling method improves the resilience against disguised jamming for a single link.

FHSS work by M. Tiloca et al. [55] is proposed in the context of IEEE 802.15.4 in TSCH mode. They created a CSPRNG based on AES-CTR. The CSPRNG output is used to execute their Secure Link Permutation (SLP) algorithm, namely a pseudo-random-permutation. SLP output pseudo-randomly determines the TSCH schedule. Also, they periodically change the TSCH schedule. They implemented their SLP algorithm in Contiki OS with TSCH, and evaluated it in TelosB IoT motes. They compared the results against a fixed non-AJ TSCH schedule. Their proposal effectively defeats a selective-jammer, with negligible energy and packet-delivery-ratio penalties.

D. Torrieri [56] discusses the concept of *maneuver keys* in the context of MTD applied to a DSSS system. The work is a high-level design of such a system: the system nodes share a given group *key* k_g that they use as the sequence for DSSS modulation. How this key is generated and distributed is not detailed. The system is reactive, i.e., Intrusion Detection System (IDS) is present in the network. Moreover, if a jammer node is detected, it will be isolated from the rest of the system by initiating a secure group re-keying that excludes it. Later, Torrieri [57] re-categorized this proposal within the cyber-defense concept of *cyber maneuvers*.

To the best of our knowledge, no multi-link wireless system AJ proposal in the literature is *proactively* resilient to an *insider-node attack*. The system proposed by Hermanns [52] is potentially resilient, but it lacks sufficient detail about the distributed mechanisms and properties of the system, AJ was evaluated for a single-link. Song et al. [54] AES-based *secure scrambling* proposal is, at the core, similar to our proposal. However, as [52], it focuses on a single link, the distributed mechanisms to generate codes for multiple users are not detailed, nor the system-wide AJ properties evaluated. Thus, under the hypothesis of an insider attacker that has knowledge of the secret parameters of one legit node, the system resilience for other nodes can not be estimated (e.g., discovering the LSFR used as input can compromise the other nodes). In Tiloca et al. [55], all nodes share the same frequency-hopping schedule: the calculation is *distributed*, but not *independent*. Inevitably, an insider attacker can efficiently jam the whole system. Finally, in Torrieri [56] all nodes share the same spreading-sequence. In case of an insider jammer, all links will be efficiently jammed, defeating the system. The system will only recover *after* some time when the IDS isolates the jammer. This is a *reactive* strategy to mitigate insider attacks. It excludes the node from the network only *after* some given process. In other words, jamming-detection is needed. Jamming-detection is a prominent field in the AJ literature and is used in reactive AJ strategies.

All works cited in this section, including our own, as-

[12] Not specified how they were generated, but probably with a CSPRNG.

[13] Which behaves functionally as a stream-cipher.

[14] *Scrambling* is also used in classical wireless literature to refer to long DSSS PN sequences, not necessarily cryptographically secure.

sume pre-shared secrets between the sender and receiver to execute AJ spread-spectrum techniques. In a real-word setting this hypothesis is not always true. In many IoT use cases, previously-unknown nodes have to bootstrap ad-hoc mesh networks. Those nodes do not share common cryptographic material. Physical-layer AJ bootstrapping is a hard problem to solve. AJ systems without pre-shared secret information are needed. This fundamental topic is called *keyless jam resistance*. We refer the reader to Kang et al. [58] for a recent survey on DSSS-based keyless AJ. And to J. Tao et al. [59], and C. Pöpper et al. [50], for pairwise and broadcast communication proposals, respectively.

7. Discussion and Future Work

In this section, we discuss security-related issues, our system proposal in the IoT context, deployment challenges, and identify future work perspectives.

7.1. Discussion: Security, IoT, and Deployment Challenges

The main novelty of our DSSS system proposal is that it uses pairwise Cryptographically Secure Pseudo-Random Spreading-Sequences generated in a *distributed* and *independent* way. We also prioritize crypto-agility; for example, our proposal can be instantiated with software-based IoT-friendly crypto primitives like ChaCha20.

The study of the Cross-Correlation properties of CSPR sequence sets proved to be fundamental to evaluate the AJ resilience of our proposal. First, we found that an in-depth study of the CC of independent uniformly distributed random binary sequences was missing in the literature; thus, we provided both an empirical statistical characterization of the CC values of large CSPR sequence sets and a probabilistic study. Then, we designed and evaluated our system in MATLAB; notably, against a power-efficient insider jammer. Finally, we linked the CC properties with the AJ resilience and characterized the AJ resilience of a generic system implementing our proposal.

Security-related issues of PRNG Sequence Sets. The AJ capabilities of a spread-spectrum wireless system depend on the *secrecy* of the sequences used. A jammer with knowledge of the spreading parameters can attack the system very power-efficiently [30, 54]. In terms of security (i.e., keeping the secrecy), the shortcomings of AJ solutions using legacy PRNG sequences are twofold. First, LSFR-based sequences (e.g., Gold codes for 3GPP-UMTS) can be brute-forced in a computationally reasonable time (Berlekamp-Massey [29, 54]); this affects the secrecy of only one sequence. Secondly, PRNG sequence sets are composed of not independent sequences (e.g., to guarantee cross-correlation). Thus, knowing one sequence leaks information about the others. This second issue is very relevant to our insider attacker setting; For example, the knowledge of one De Bruijn low-CC code will ease the task of breaking the other codes in the set. The first issue can be addressed by using non-LSFR codes, and has been explored in the bibliography (e.g., De Bruijn, AES-based codes). The second one can only be addressed by sets where all the sequences are independent of each other (e.g., no cross-correlation constraints). CSPR codes generated independently addresses both weaknesses. To the best of our knowledge, our work is the first one to propose sequence sets for WCS composed of independent sequences, which we called CSPR Sequence Sets.

Non-security impacts of CSPR Sequence Sets. Our proposed CSPR-based sequence generation process prioritizes as design factors cryptographically secure randomness and independence of sequences. This has an impact in the Cross-Correlation (CC) of the sequences in the sets. There are no hard-guarantees about CC max values. We refer the reader to Sec. 4.6 about the implications in comparison to other families of sequences. To complete the CC discussion, we add that one of the main contributions of our work is the study in Sec. 4 of the CC properties of CSPR sequences sets that was lacking in the bibliography. We have not fully-studied the consequences they have for real systems in the current work (i.e., we focused on their AJ capabilities and evaluated for an AWGN-model). However, their statistical properties look promising, specially for low-throughput and systems with many nodes (or with potential rotation of codes per user as in our proposal). For example, the mean value ($\sqrt{2/\pi L}$) and p.m.f. of the normalized cross-correlation as a function of the sequence length L, can be used to have an estimation of the performance of large IoT systems implementing MTD-dynamic CSPR sequences. Also, because of the perpetual rotation of the used sequences, the CC statistics properties are relevant even for a single node. Regarding the performance of the system when not under jamming, Hermanns [52] studied AES-based sequences for systems with 1 to 40 nodes and found that they perform as Gold Codes.

Another trade-off could be the computational cost of generating CSPR sequences as compared to LSFR-based solutions. Advances in hardware and software allow for suitable implementations of CSPRNG on IoT devices that make this not being an issue. Most IoT SoCs have AES Hardware Modules, and software-based solutions -like ChaCha20- are fast on IoT devices [46].

Relevance of Proposal for IoT Systems. The AJ resilience against an insider jammer can only be measured in terms of probability. The pseudo-randomness and independence of sequences at the core of our system design are the main causes. In contrast, classical wireless systems precisely determine many properties *a priori*, i.e., with probability 1. For example, maximal CC or max-bounds for multi-hop latency. However, this lack of hard-values certainty is not a big drawback for the MTD IoT systems we target. A large number of nodes and MTD-inspired rotation of sequences over time are both properties that make any particular system to converge statistically to the CC -and thus AJ- properties we studied. Furthermore, the cyber-defense objective in our IoT setting is not to protect a single (or limited number of) primary wireless targets, like a satellite link, a radar system, or cellphone-users. Instead, the *IoT system as a whole* is the target to defend. In other words, we care about the *service*

the IoT system provides, and not the individual IoT nodes. For example, we can design a system to provide a given service, even if only 10% of the IoT nodes (or any given node only 10% of the time) will be resilient to a +25dB powerful jammer. In constrained-node IoT networks "strength lies in (big) numbers".

Key deployment challenges. A real-world implementation of our proposal will have to deal with the synchronization of spreading-sequences for signal demodulation at the RX-node. This problem can be solved because CSPR sequences have good auto-correlation properties [21]. Also, the *bootstrapping* of the system (i.e., the pre-shared keys and time synch.) is not a trivial problem. In this respect, keyless jam resistance [58] techniques can be used to execute higher-layer communication protocols (e.g., key establishment). Another point to be discussed is the availability of DSSS technologies in real-world IoT systems. IoT hardware uses mostly narrow-band technologies, where FHSS techniques are dominant. IEEE 802.15.4 has a DSSS mode, but the spreading-sequences have length $L = 16$, which is not long enough for robust AJ. However, Software-Defined-Radio (SDR) technology has yet a bigger role to play in the future of IoT [60]. Cost-affordable SDR technologies will add to the synergy of new-services enabled by the IoT. In this context, alternating between DSSS, FHSS, or Long-Range radio will be a matter of executing some lines of computer code in already-deployed IoT nodes.

7.2. Future Work

Future work perspectives include the definition of other adversarial settings and a comparative study of AJ proposals in these settings. Those adversarial settings can include dynamic attacker-system interactions. For those scenarios, our proposal has the elements needed to design and implement an adaptive AJ defense strategy. We are working on an SDR platform implementation of our proposal that will allow us to explore IoT real-world use cases.

8. Conclusion

IoT systems are inherently exposed to jamming. State-of-the-art IoT systems do not have AJ properties as one of their main objectives. Moreover, insider attacks are a real threat in the heterogeneous IoT ecosystem. Inspired by the MTD paradigm, we provided an IoT-suitable DSSS AJ solution proactively resilient to insider attacks. The main novelty of our solution is the use of spreading sequences independently generated with CSPRNGs. We evaluated the AJ capabilities of our system proposal using MATLAB. Inspired by an insider attack, we exposed the system to a power-efficient *insider smart jammer*. The experimental results validated the insider jammer resilience claim of our proposal. They are explained by the cross-correlation properties of CSPR sequence sets, for which we provided an in-depth statistical and probabilistic study that was missing in the literature.

Acknowledgments

Kindly thanks for their precious time to Manuel Lagos, Alexandre Marquet, Patrick Maillé, and Xavier Lagrange. To the neccesary support of ASSET Project (German-French Academy for the Industry of the Future), and the Contrat Plan Etat-Région.

A. Unbreakable Spreading Sequences

The smart jammer's hypothesis *the jammer can not gain knowledge of the Spreading Sequence (SS) of another node of the system* limits the capabilities of the jammer. This assumption is motivated by the dynamic nature of the attacker-system relationship, and by imposing it, we are simplifying this dynamism. If we assume that the jammer can gain knowledge of an unknown SS, we have to estimate a Δtime= t_{attack_ss} needed for it. From an attacked node perspective, once the attacker knows the sequence, we are *defeated* (BER $= 0.5$ for JSR$\approx$0). If we use the moving target defense (MTD) aspect of our proposed system, we can mitigate this attack: the MTD system has to change the SS of the nodes with a periodicity $T_{ss\_movement} < t_{attack\_ss}$. We simplify the attacker model with two possible states: either knows the SS of a node ($t \geq t_{attack\_ss}$) or does not ($t < t_{attack\_ss}$). As stated before, in our proposal breaking one SS does not imply breaking the other SS. This is due to the independence in the generation of the CSPR sequences, knowing one sequence does not leak any information about other SS (unlike other PR proposals in the literature). From a system perspective, it is very relevant to study the case in which the attacker has gained knowledge of *one node* SS (or equivalently, using a uniformly random SS), and measure the impact over the BER of the system excluding the compromised node, as done in Section 5.4.

B. Asymptotic AJ Evaluation

A smart jammer affects each node differently, and the way it affects each node is related to the Normalized Cross-Correlation (NCC) between the jammer and node sequences (Eq. 5). We use the results from Sec. 5.3 and Sec. 5.4 to calculate percentiles for nodes with $BER \leq 0.1$ for the Smart Jammer (SJ), and BroadBand-Noise (BBN) jammer scenarios. We present the results in Fig. 8. The ECDF of the NCC entirely determines the SJ case. Some counter-intuitive results happen on the P_{10} for $L \in \{2^9, 768, 2^{10}\}$ due to the discrete nature of the NCC values. Aside from that, for a given percentile we observe that if we double the length L we gain $\approx +1.5dB$ ($\sqrt{2}$ in linear terms) in jammer resilience.

References

[1] K. L. Lueth, State of the iot 2018: Number of iot devices now at 7b, accessed: 2019-04-18 (2018) [cited 14-01-2019].
URL https://iot-analytics.com/state-of-the-iot-update-q1-q2-2018-number-of-iot-devices-now-7b/

[2] C. Bormann, M. Ersue, A. Keränen, Terminology for Constrained-Node Networks, RFC 7228 (May 2014). doi:10.17487/RFC7228.

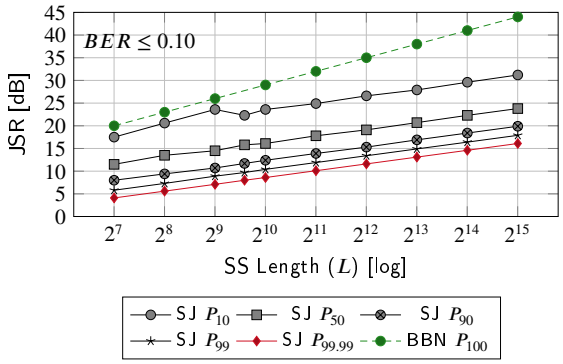


Figure 8: Percentiles of nodes with $BER \leq 0.1$ for a Jammer with power JSR (dB), as a function of the SS length L.

[3] C. Kolias, G. Kambourakis, A. Stavrou, J. Voas, Ddos in the iot: Mirai and other botnets, Computer 50 (7) (2017) 80–84.

[4] E. Ronen, et al., Iot goes nuclear: Creating a zigbee chain reaction, in: 2017 IEEE Symposium on Security and Privacy (SP), IEEE, 2017.

[5] M. A. Khan, et al., Iot security: Review, blockchain solutions, and open challenges, Future Generation Computer Systems (2018).

[6] A. Ghosh, et al., Moving target defense co-chair's report-national cyber leap year summit 2009, TR, Federal NITRD Program (2009).

[7] G.-l. Cai, et al., Moving target defense: state of the art and characteristics, Frontiers of Information Technology & Electronic Eng. (2016).

[8] S. Jajodia, et al., Moving target defense: creating asymmetric uncertainty for cyber threats, Springer Science & Business Media, 2011.

[9] H. Okhravi, et al., Finding focus in the blur of moving-target techniques, IEEE Security & Privacy 12 (2014).

[10] B. C. Ward, et al., Survey of cyber moving targets second edition, Tech. rep., MIT Lincoln Laboratory Lexington United States (2018).

[11] C. E. Shannon, A mathematical theory of communication, Bell tech. journal (1948).

[12] S. Stojanovski, et al., Efficient attacks in industrial wireless sensor networks, in: International Conf. on ICT Innovations, Springer, 2014.

[13] P. G. Flikkema, Spread-spectrum techniques for wireless communication, IEEE Signal Processing Magazine 14 (3) (1997).

[14] M. K. Simon, et al., Spread spectrum communications handbook, MG-Hill, 1997.

[15] IEEE, 802.15.4e Standard for Local and metropolitan area networks. Part15.4: Low-Rate Wireless Personal Area Networks(LR-WPANs)A.1:MAC sublayer (2012).

[16] M. Knight, B. Seeber, Decoding lora: Realizing a modern lpwan with sdr, in: Proceedings of the GNU Radio Conference, 2016.

[17] E. Aras, et al., Exploring the security vulnerabilities of lora, in: 2017 3rd IEEE International Conference on Cybernetics, IEEE, 2017.

[18] X. Cheng, J. Shi, M. Sha, Cracking the channel hopping sequences in ieee 802.15.4e-based industrial tsch networks, 4th IoTDI 2019 (2019).

[19] S. W. Golomb, G. Gong, Signal design for good correlation: for wireless communication, cryptography, and radar, 2005.

[20] J. M. Velazquez-Gutierrez, C. Vargas-Rosales, Sequence sets in wireless communication systems: A survey, IEEE Communications Surveys & Tutorials 19 (2) (2017).

[21] H. D. Schotten, H. D. Lüke, On the search for low correlated binary sequences, AEU-Inte. Journal of Electronics and Comm. (2005).

[22] M. Golay, The merit factor of long low autocorrelation binary sequences (corresp.), IEEE Transactions on Information Theory 28 (1982).

[23] D. V. Sarwate, M. B. Pursley, Crosscorrelation properties of pseudorandom and related sequences, Proceedings of the IEEE 68 (5) (1980).

[24] H.-J. Zepernick, A. Finger, Pseudo random signal processing: theory and application, John Wiley & Sons, 2005.

[25] S. Spinsante, et al., Binary de bruijn sequences for ds-cdma systems: analysis and results, EURASIP Journal on Wireless Communications and Networking (2011).

[26] V. M. Sidelnikov, On mutual correlation of sequences, Probl. Kybern. 24 (1971).

[27] L. Welch, Lower bounds on the maximum cross correlation of signals (corresp.), IEEE Transactions on Information theory 20 (1974).

[28] S. Golomb, Shift Register Sequences (3rd Rev. Edition), World Scientific, 2017.

[29] J. Massey, Shift-register synthesis and bch decoding, IEEE Transactions on Information Theory (1969).

[30] S. Amuru, et al., Optimal jamming against digital modulation, IEEE Transactions on Information Forensics and Security 10 (2015).

[31] I. B. Damgård, On the randomness of legendre and jacobi sequences, in: Conf. on the Theory and App. of Cryptography, Springer, 1988.

[32] Z. Chen, X. Du, G. Xiao, Sequences related to legendre/jacobi sequences, Information Sciences 177 (2007).

[33] S. Spinsante, et al., De bruijn binary sequences and spread spectrum applications: A marriage possible?, IEEE Aerospace and Electronic Systems Magazine (2013).

[34] C. Warty, et al., De bruijn sequences as secure spreading codes for wireless communications, in: ICACCI, IEEE, 2013.

[35] T. Kohda, et al., Statistics of chaotic binary sequences, IEEE Transactions on information theory (1997).

[36] G. Mazzini, et al., Chaotic complex spreading sequences for asynchronous ds-cdma. i. system modeling and results, IEEE Trans. on Circuits and Systems (1997).

[37] C. Tse, et al., Chaos-based digital communication systems, Operating Principles, Analysis Methods and Performance Evaluation (2003).

[38] A. Akhavan, et al., Cryptanalysis of "an improvement over an image encryption method based on total shuffling", Optics Comm. (2015).

[39] C. Li, Cracking a hierarchical chaotic image encryption algorithm based on permutation, Signal Processing 118 (2016) 203–210.

[40] E. Barker, et al., Recommendation for random number generation using deterministic random bit generators, NIST SP 800-90A R1 (2015).

[41] L. E. Bassham, et al., A statistical test suite for random and pseudorandom number generators for cryptographic applications, NIST SP 800-22 Rev 1a (2010).

[42] R. A. Rueppel, Analysis and Design of Stream Ciphers, Springer-Verlag, 1986. arXiv:arXiv:1011.1669v3.

[43] D. J. Bernstein, Chacha, a variant of salsa20, in: SASC, 2008.

[44] Y. Nir, ChaCha20, Poly1305, and Their Use in the Internet Key Exchange Protocol (IKE) and IPsec, RFC 7634 (Aug. 2015).

[45] A. Langley, et al., ChaCha20-Poly1305 Cipher Suites for Transport Layer Security (TLS), RFC 7905 (Jun. 2016). doi:10.17487/RFC7905.

[46] F. De Santis, A. Schauer, G. Sigl, Chacha20-poly1305 authenticated encryption for high-speed embedded iot applications, in: Proceedings of the Conference on Design, Automation & Test in Europe, European Design and Automation Association, 2017, pp. 692–697.

[47] H. G. Tucker, A generalization of the glivenko-cantelli theorem, The Annals of Mathematical Statistics 30 (3) (1959) 828–830.

[48] R. Poisel, Jamming techniques, in: Modern Communications Jamming Principles and Techniques, Artech House, 2011, Ch. 8.

[49] P. Benprom, et al., Analysis of convolutional coded direct sequence spread spectrum cdma system with a bpsk jamming signal, in: 8th ECTI, IEEE, 2011.

[50] C. Pöpper, Jamming-resistant broadcast communication without shared keys., USENIX security Symposium (2009).

[51] L. Kuipers, et al., Unifom distribution in topological groups: Convolution of sequences, in: Uniform distribution of sequences, John Wiley & Sons, 1974, Ch. 4.

[52] F. Hermanns, Secure and robust tactical communications based on code-hopping cdma (ch-cdma), NATO/OTAN, Germany, Report No. RTO-MP-IST-083 (2008).

[53] F. Hermanns, Protected spread spectrum signal transmission system for multiple-access messages uses pseudo-random sequences as expansion codes, german Patent DE102004013884B4 (2004).

[54] T. Song, K. Zhou, T. Li, Cdma system design and capacity analysis under disguised jamming, IEEE Transactions on Information Forensics and Security 11 (11) (2016) 2487–2498.

[55] M. Tiloca, et al., Dish: Distributed shuffling against selective jamming attack in ieee 802.15.4e tsch networks, ACM TOSN (2018).

[56] D. Torrieri, et al., Cyber maneuver against external adversaries and compromised nodes, in: Moving Target Defense II, Springer, 2013.
[57] D. Torrieri, Cyber maneuvers and maneuver keys, in: 2014 IEEE Military Communications Conference, IEEE, 2014, pp. 262–267.
[58] T. Kang, et al., A survey of security mechanisms with direct sequence spread spectrum signals, Journal of Comp. Science and Eng. (2013).
[59] T. Jin, et al., Zero pre-shared secret key establishment in the presence of jammers, in: ACM MobiHoc '19, 2009.
[60] A. A. Khan, et al., Cognitive-radio-based internet of things: Applications, architectures, spectrum related functionalities, and future research directions, IEEE wireless communications 24 (3) (2017).