Pseudorandom Sequences in Spread-Spectrum Communications Generated by Cellular Automata

F.C. Ordaz-Salazar[1], J.S. González-Salas*[1], E. Campos-Cantón[2], H.C. Rosu[2]

[1] Universidad Politécnica de San Luis Potosí
San Luis Potosí, S.L.P. México
*jsgs100573@hotmail.com
[2] Instituto Potosino de Investigación Científica y Tecnológica
San Luis Potosí, S.L.P., México

ABSTRACT
Dynamical systems methods have been recently used in spread-spectrum digital communication systems. The expansion of the spectrum using a pseudorandom sequence with a higher frequency than the information signal is the key feature for its robustness against the signal traveling interference through the channel. In this work, we propose to generate pseudorandom sequences by employing cellular automata and we check these sequences have the necessary properties which are required in modern communication systems. The computed sequences obtained by the cellular automata are tested in a quadrature phase shift keying (QPSK) spread-spectrum communication system. The efficiency of the system is analyzed by computing the bit error rate under different signal to noise ratio conditions. These results are compared with systems that employ Golden code and other typical pseudorandom sequences.

Keywords: pseudorandom sequences, cellular automata, spread spectrum.

RESUMEN
Métodos de sistemas dinámicos se han estado utilizando recientemente en sistemas digitales de comunicación de espectro expandido. La expansión del espectro mediante el uso de una secuencia pseudo-aleatoria de frecuencia mayor que la frecuencia de la señal de información es la clave que lo caracteriza para su robustez en contra de la interferencia de la señal en el canal. En este trabajo se propone generar secuencias pseudo-aleatorias mediante el uso de autómatas celulares y verificar que esta secuencias tengan las propiedades necesarias que se requieren para sistemas de comunicación modernos. Las secuencias generadas con autómatas celulares son probadas en un sistema de espectro expandido que utiliza modulación por cuadratura de fase QPSK. La eficiencia del sistema es analizada mediante el cálculo de la taza de error de bit bajo diferentes condiciones de la razón señal a ruido. Estos resultados son comparados con sistemas que emplean otras secuencias pseudo-aleatorias y secuencias *Golden code*.

1. Introduction

In the realm of digital communication, the spread-spectrum (SS) communication system has become the preferred system for wireless communications [1]. It features a robustness effect against intentional interference by other users that share the same channel, robustness against self-interference produced by multipath effects and signals on the channel which are difficult to demodulate by other receivers than the intended ones. The SS communication system is desirable to be used in cellular phone technologies, cordless phones, WLAN, Bluetooth and GPS. The pseudorandom number (PN) sequence, which spreads the signal, is the key element to get these good properties of the SS system. This is conventionally generated by a linear feedback shift register (LFSR) which generates known sequences like the Gold and Kasami sequences and the Golden code ones [2,3].

In recent research in the area of SS systems, a model of a direct sequence spread spectrum (DSSS) for voice and data transmission [4] was designed; its authors showed in numerical simulations that this model works in a satisfactory way. In turn, the authors of [5] designed a new pseudorandom sequence generator for multichannel DSSS systems in which Kasami sequences and Golden code are used to reduce the total of distortion, including the distortion due to

intersymbol interference. Other applications for SS systems are described in [6] where it is given an overview of how SS systems are used to cover secret messages in digital images by using PSK techniques.

With the goal to improve SS communication, some research has been oriented towards the use of dynamical systems for secure communication [7,8] and for the generation of PN sequences. Following the second line of research, it began with Heidari-Betani and McGillem [9], who proposed the use of chaotic sequences instead of pseudorandom (PN) sequences. In turn, Drake and Williams [10] studied a dynamical system based on the sawtooth map to produce pseudo-chaotic sequences applied to a direct-sequence SS communication system. Later on, Wang and Hu [11] proposed the logistic map to generate the PN sequence while Leon et al. [12] exploited the quantization of chaotic maps. Besides, there have been some efforts to analyze the properties of this kind of sequences, for example Xia and collaborators [13] analyzed chaotic binary sequences and their application to code division multiple access (CDMA) systems. Finally, the characterization of chaotic sequences, for use in spread-spectrum communication schemes, was studied by Micco et al. [14].

In this work, we propose the use of discrete dynamical systems such as pseudorandom number generator (PNG). In particular, we propose the use of cellular automata, which offer some advantages versus other dynamical systems such as ease of hardware implementation, high speed sequence generation, and the benefits of the generated sequence.

2. Elementary cellular automata (ECA)

An elementary cellular automaton is a discrete dynamical system which consists of a one-dimensional lattice of cells, an update rule which involves a given cell and its next two adjacent cells, and a binary alphabet. A one-dimensional circular lattice of N cells labeled by $i \in \{0,\ldots,N-1\}$, $a_i \in Z_2$, and with the updating rule:

$$a_i^{t+1} = a_{i-1}^t \text{ XOR } (a_i^t \text{ OR } a_{i+1}^t) \qquad (1)$$

where a_i^t is the value of cell i at time t. Given an initial state on the lattice, the cells evolve according to the updating rule which generates the evolution of the system as shown in Figure 1. In general, the updating rule is homogeneous for all cells and it has been found that even the simplest rules can generate very complex behaviors [16].

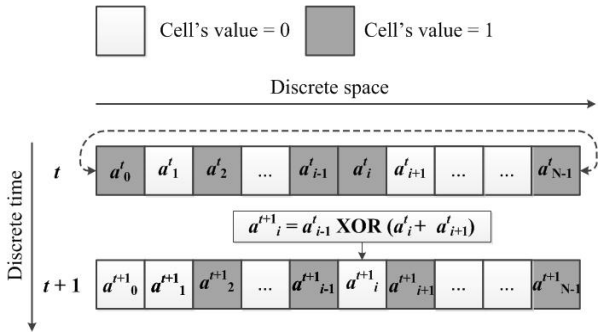

Figure 1. Example of an ECA evolution.

The names of all possible rules of ECA are based on the following argument: since a given ECA neighborhood contains three cells, each with alphabet Z_2, then 2^3 possible neighborhoods exist, and since each neighborhood generates a result in Z_2, then 2^{2^3} possible rules exist. The binary number produced by the results of the application of each rule to every neighborhood converted to decimal number represents the name of the rule, see the table in Figure 2 [15].

Wolfram reported four possible behaviors of the ECA evolutions. The first one is very simple; the evolution converges to a homogeneous stage (as in rules 0, 4, 16, 32, 36, 48, 54, 60, and 62). In the second type of behavior, the evolution converges to a stage in which a set of simple or periodic stable structures exists (as in rules 8, 24, 40, 56 and 58). The third behavior evolves to a stage with chaotic patterns (as in rules 2, 6, 10, 12, 14, 18, 22, 26, 28, 30, 34, 38, 42, 44, 46, and 50). Finally, the fourth class converges to stages with complex structures which are fixed in space for a long time or they can move through space [17].

We are interested in ECA with chaotic behavior because they could have random properties. It is known Rule 30 (R30) is one of the most studied rules in the area of cellular automata applied to PN

sequences [16]. It has better randomness properties because of which it generates larger sequences than the other rules with chaotic behavior [15]. We choose R30 as a PNG for our DSSS communication system because larger PN sequences give more security when hiding information to our communication system.

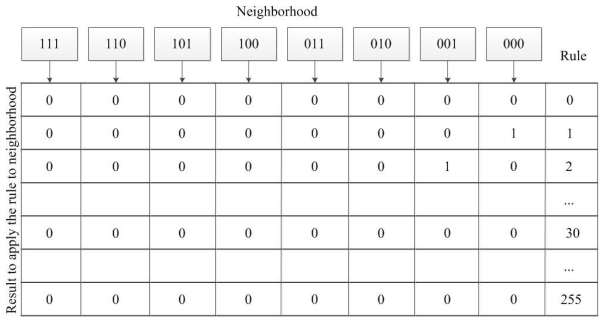

Figure 2. Names of the cellular automata rules in the last column according to Wolfram.

2.1 Proportion of sequences passing a test

To establish a general conclusion, NIST adopts two criteria for each test.

- The first one refers to the calculation of the proportion of sequences that passes a statistical test. To accept proportion ρ, it must be in the following range of values:
- $$\sigma\text{-R}\leq\rho\leq\sigma\text{+R} \qquad (2)$$
- The second criterion checks the uniformity of the sequences by analyzing the distribution of the *P - values*. By using the χ^2 distribution, the test computes a value corresponding to the Goodness-of-Fit Distributional Test called the P - $value_T$. If the latter is equal to 0.0001 or greater than the distribution of the P – values is considered as uniform [18].

Table 1 displays the results obtained by applying the sixteen tests applied to 1000 sequences generated with an ECA R30 of 64-bit size. Except for the approximate entropy and the universal tests requiring longer sequences than those used in the other tests [18], each sequence has a length of 30000 data. Sequences of 40000 and 390000 data are used for the approximate entropy test and universal test, respectively. Table 1 shows that all sixteen tests pass both criteria of proportion and uniformity, unless the random excursion test, which does not fulfill the uniformity criterion. Therefore, according to this evidence obtained from the NIST test, we can conclude that ECA R30 is a good PNG

Test	Proportion	Uniformity	Success
Frequency	0.994	0.994	Yes
Block Frequency	0.993	0.2088	Yes
Cumulative Sums	0.0375	0.994	Yes
Cumulative Sums B	0.1068	0.993	Yes
Run	0.988	0.1001	Yes
Longest Run	0.996	0.0743	Yes
Binary Matrix Rank	0.99	0.419	Yes
Discrete Fourier Transform	0.986	0.00013	Yes
Non-overlapping Template Matching	0.988	0.3804	Yes
Overlapping Template Matching	0.993	0.5769	Yes
Universal	1	0.9357	Yes
Random Excursions	1	—	Yes
Entropy	0.99	0.0589	Yes
Serial	0.991	0.1529	Yes
Serial	0.992	0.4064	Yes
Linear Complexity	0.984	0.0318	Yes

Table 1. NIST suite results for one thousand 64-bit sequences of ECA R30.

3. Test for spread spectrum

In this section, we perform numerical tests for the cellular automata pseudorandom sequences which give us confidence that these sequences are useful for spread spectrum.

3.1 Auto-correlation and cross-correlation

Consider two binary sequences of N data $\{a\}=\{a_0, a_1,a_2,\cdots,a_n\}$ and $\{b\}=\{b_0, b_1,b_2,\cdots,b_n\}$. The cross-correlation function between these two sequences is calculated as follows:

$$L_{ab}(\tau)=\frac{1}{R_{ab(0)}}\sum_{n=0}^{N-1} a_n \cdot b^{*}_{n+\tau}, \quad (3)$$

where b_n^{*} denotes the complex conjugate of b_n. If b=a, then $L_{aa}(\tau)$ is the autocorrelation of sequence {a}. As when studying a PN generator, then it is necessary to analyze properties of correlation in great number sequences. One way is to give averages of $L(\tau)$ for each shift time τ. Then, to calculate the cross-correlation of *M* sequences $\{a_i^k\}$ with *M* sequences $\{b_i^k\}$, respectively, i.e., $L_{a^k b^k}(\tau)$ for $1 \le k \le M$, we calculate its average as follows:

$$\bar{L}_M(\tau)=\frac{1}{M}\sum_{k=1}^{k=M} L_{a^k b^k}(\tau). \quad (4)$$

First, we study the auto-correlation in sequences generated with an ECA R30. In particular, we calculate the average $\bar{L}_M(\tau)$ of the autocorrelation based on 1000 PN sequences, where each PN sequence is initialized with a different seed and has a length of 30000 data. Besides, each seed is randomly selected. As it was expected for PN sequences, Figure 3 (a) shows all the values of random alternation around the value 0.0, except for $\bar{L}(0)=1.0$. Afterwards, we calculate the cross-correlation between 1000 pairs of ECA R30 sequences initialized with different seeds (as above, each seed is randomly selected). Figure 3 (b) shows the average $\bar{L}_M(\tau)$ which has good properties to be used as PN sequences in CDMA systems because each user is assigned to a particular PN sequence.; therefore, in order to access the channel, the level of interference is decreased when these sequences are orthogonal, i.e., when the sequences have cross-correlation with values close to zero.

3.2 Balance of probability

Another useful test for PN sequences employed in spread spectrum is balance of probability B which is defined as follows:

$$B(n)=(p-q)/n \quad (5)$$

which are used to calculate these occurrences. Figure 4 illustrates the average of 1000 sequences generated with cellular automata in function of size n. As it was expected in PN sequences, B(n) is zero for almost every n.

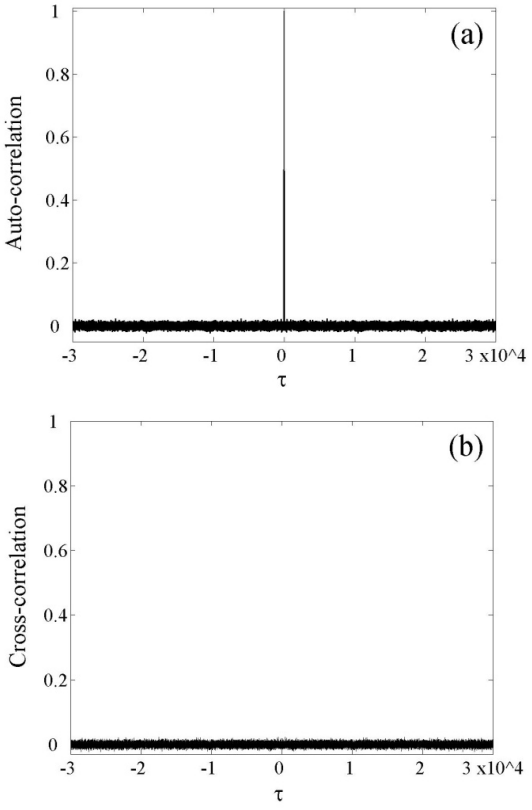

Figure 3. Autocorrelation (a) autocorrelation and (b) cross-correlation of 1000 sequences generated with an ECA R30.

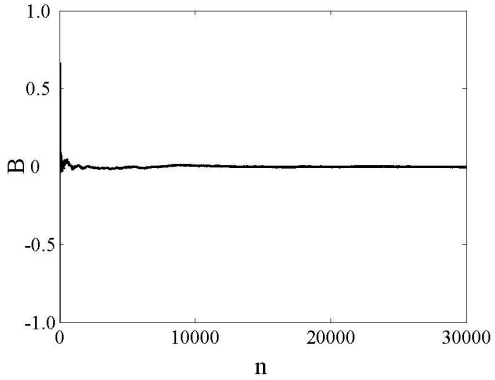

Figure 4. Average of balance of probability of 1000 sequences generated with an ECA R30.

4. SS Communication system

Figure 5 shows the block diagram of the direct sequence SS system that we used to apply the PN sequences generated by a CA. It has three component blocks: the transmitter, the channel, and the receiver. The digitalized information signal is the input of the transmitter, which includes a PNG, an XOR logic function, an encoder, and a modulator. Noise is included in the channel by adding it to the transmitted sequence. The receiver includes a demodulator, a decoder, a PNG, and an XOR logic function. In the simulations of this communication system, we use the QPSK modulation method for data transmission through the channel.

Let s_i be the binary sequence with the information signal which flows with a bit rate of R_a. A SS system makes signal s_i on the channel to be hidden as a random signal. This is achieved by adding a PN sequence of a bi rate R_b to the information signal, where the ratio between R_b and R_a must be a number bigger than one, i.e., $R_b/ R_a \geq 1$.

In Figure 5, the PNG block in the transmitter represents the binary pseudorandom number generator that provides sequence a_i which is used to be added to the information signal. This operation is represented in the transmitter block by the XOR block, which generates the extended sequence $d_i=a_i \otimes s_i$ where $\otimes$ represents the XOR operation.

In order to decrease the errors during the communication process, the SS system includes the Gray codification before the modulation in the transmitter and after the demodulation in the receiver. This means that we apply the Gray code to the extended sequence d_i. The extended sequence which was codified by the Gray code is called *sequence* g_i, and this sequence is modulated by using QPSK modulation. After the modulation, the carrier signal is put on the channel where the noise is added to it.

Once the signal has passed the channel, the receptor demodulates the received signal by using a QPSK demodulator. As a result, the receptor obtains the demodulated sequence D_i, which is a binary sequence of 0s and 1s. Afterwards, the QPSK modulation is applied to the D_i one generating the Gray decoded sequence G_i. To regain the signal information, G_i is added to the PN sequence (generated in the PNG block of the receptor) by using the XOR logic function. To achieve a correct rebuilding of the information signal, the PN sequence in the transmitter must be identical to the PN sequence in the receiver, i.e., in a simple way, the PN sequence generated in the transmitter is the same as the PN sequence used in the receiver. One of the traditional procedures to generate PN sequences is by using Golden codes. However, in this work, the PN sequences are generated by cellular automata as one more alternative.

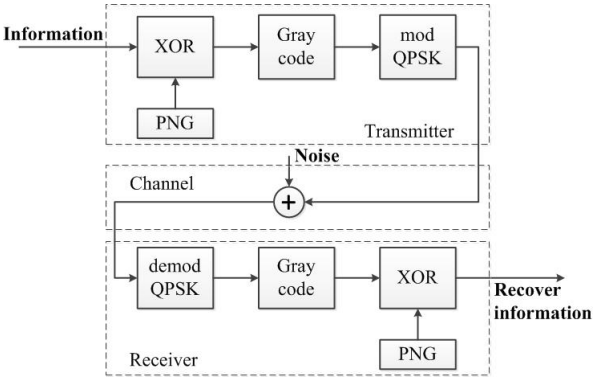

Figure 5.-Block diagram of a spread-spectrum system using the QPSK modulation method.

4.1 Numerical simulations for SS systems

In this section, we describe the numerical computations which we performed to simulate the SS system with QPSK modulation and cellular automata as PNG.

For the transmitted information signal, we used a human voice signal which is plotted in Figure 6 while its Fourier transform is plotted in Figure 7. This human voice signal was obtained from a codified file with a WAV format. As a first step, we convert the WAV file to a file in binary format where each datum of the WAV file is symbolized by a sixteen binary word. Let w_i be a datum of the WAV file, as WAV data are numbers between the

interval [-1,1], to transform each datum in a entire number in the interval $[0,2^{16}-1]$, we calculate $(w_i+1)*2^{15}$ for each value of the WAV file. Then each entire decimal number is converted to its corresponding binary sixteen-bit number. The sequence of these binary numbers in a serial form will be information signal s_i that we used to simulate in the DSSS transmission.

The PN sequence a_i (i=0,···, N-1) is generated with an ECA R30 with a 64-bit size whose initial state is produced with function randsrc. As we want to simulate a PN sequence with a bit speed 8 times greater than the bit speed of the information signal, then each bit of this signal is replicated 8 times, i.e., $R_a/R_b=8$. Afterwards, d_i is formed by applying an XOR operation between the replicated signal and the PN sequence. In order to show the spread-spectrum method in the generation of sequence d_i and corroborate how the spectrum of the voice signal is scattered, we compared the spectrums of the voice signal and the spectrum of the respective analogue of sequence d_i. First, we divide the sequence d_i in 16-bit sections, each section is converted to its respective decimal number δ_i . Every decimal number is transformed to a number in the interval [-1,1] by calculating $2(\delta_i/2^{16}-0.5)$. For data in the interval [-1,1], we obtain the FFT which is shown in Figure 8. As we can see, this spectrum is much more widespread than the spectrum in Figure 7.

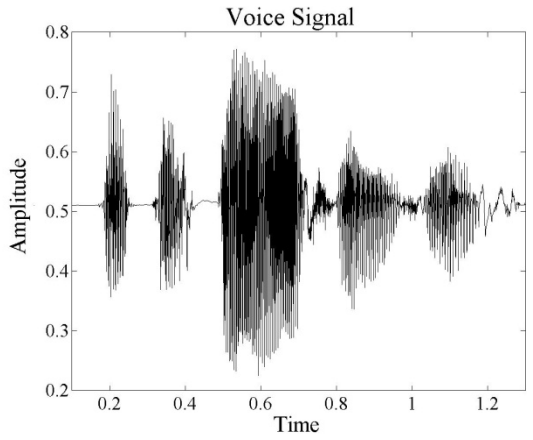

Figure 6. Information signal obtained from a human voice.

The Gray code is applied to each 2-bit word $\{d_{2k},d_{2k+1}\}$ of the extended sequence, where $0\leq k\leq \lfloor N/2\rfloor-1$ and $\lfloor x\rfloor$ is the floor function of x. Each Gray-coded word is represented by its equivalent in decimal base which forms a new sequence m_k with symbols in $Z_4=\{0,1,2,3\}$.

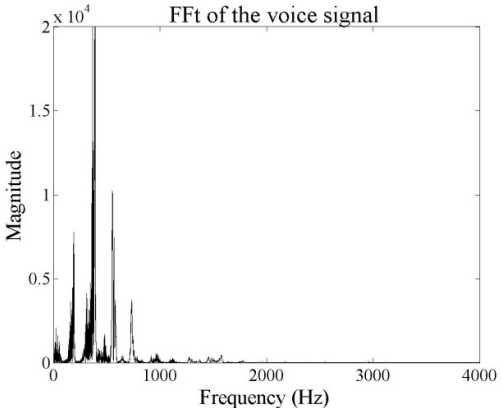

Figure 7. FFT of the information signal obtained from a voice human.

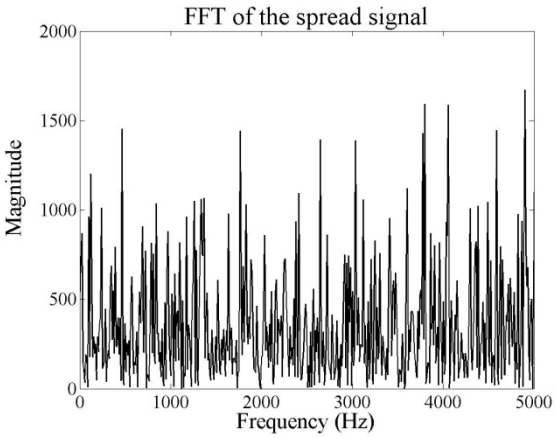

Figure 8. FFT of the human voice signal spread with a cellular automata R30.

Afterwards, sequence m_k is modulated by QSPK modulation as follows:

$$\cos\left(w_c t+\frac{\pi(1+2m_k)}{4}\right) \quad , \qquad (6),$$

where w_c is the carrier frequency. In order to achieve a good communication process it is

necessary that $w_c \geq 2\pi R_b$. For numerical simulations, we select arbitrarily the carrier frequency as one multiple of the PN sequence rate, i.e., $w_c=2\pi R_b$. For the channel, the noise is simulated in an additive way, $C_\kappa(t)=C(t)+\kappa\eta(t)$, where η is a uniformly distributed variable in the interval [-1, 1] and κ is its amplitude, whereas in the receptor, the received sequence from the channel is demodulated by passing the sequence through a low-pass filter h(t):

$$\rho_I(t)=C_\kappa(t)\cos(w_c(t))*h(t), \quad (7)$$

$$\rho_Q(t)=C_\kappa(t)\sin(w_c(t))*h(t). \quad (8)$$

The demodulated sequence D_i is obtained by determining the binary value that corresponds to each bit of the demodulated signal by using the following criterion: if $\rho_i(t)>0$, then $D_{2k}=1$, otherwise $D_{2k}=0$. If $\rho_Q(t)>1$ then $D_{2k+1}=1$ otherwise $D_{2k+1}=0$. Afterwards, we apply the Gray code to each 2-bit word $\{D_{2k},D_{2k+1}\}$ of the demodulated sequence. Next, we calculate the XOR operation between the Gray decoded sequence and the same PN sequence applied in the transmitter which gives as a result the G_i. Because 8 bits represent a datum of the original signal information sequence we separate the sequence obtained after the XOR operation in 8-bit blocks. To obtain the *n*th bit of the recovered signal information sequence $\bar{s}(n)$, the average of one block is calculated and rounded to the nearest binary integer (0 or 1) as follows:

$$\bar{s}(n)=\frac{1}{8}\sum_{k=0}^{7} G_{8n+k}, \quad (9)$$

To analyze the results, the signal information and the recovered signal information are compared by using the bit error rate (BER) as a function of the signal-to-noise ratio κ. If a bit $\bar{s}(n)\neq s(n)$, then BER=BER+1/N. With the goal of comparing the efficiency of SS communication systems using an ECA R30 PN generator with other PN generators, we perform the same numerical simulations of the SS system as before but using the Golden code and randsrc Matlab function as PN generators. Figure 9 shows BER versus SNR of the SS communication system using the Matlab number random generator, Golden sequences, cellular automata rule 30 as PNGs. By looking at the graphical results plotted in Figure 9, one can infer that the SS system using PNG based on cellular automata has a good behavior, similar to other SS systems using different PN generators.

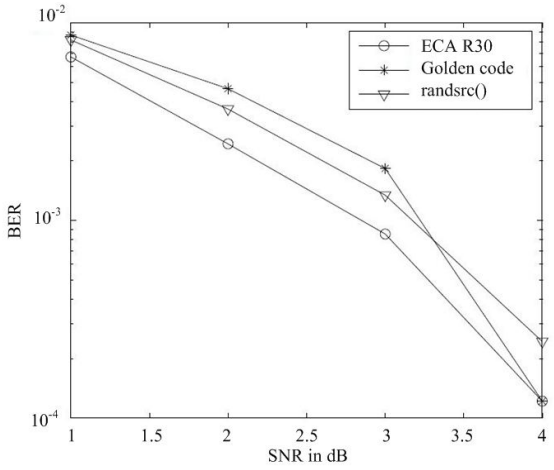

Figure 9. Bit error rate versus signal-to-noise ratio in dB for different types of PNGs.

5. Conclusions

Cellular automata are good PN generators for SS communication systems because of several useful features such as the ease of hardware implementation, the high speed of sequence generation, and their random properties. One particular characteristic of PNGs based on cellular automata is that they can generate binary data streams obtained from different sequences in contrast with conventional PNG like Golden code and shift register which generate random data streams from a single very large sequence. The NIST criteria show that ECA R30 PNGs have good random properties. Besides, these sequences have good properties for CDMA systems. Finally, from the numerical evidence of the SS communication systems based on CA-PNGs, we conclude that they are as efficient as the SS systems based on the Golden code or other random sequences.

In numerical simulations, the same PNG is used in both the transmitter and receiver. However, in real physical systems it is necessary to assure the synchronization before the transmission of the information signal, i.e., the PNGs in the transmitter and in the receiver must be synchronized. Actually,

SS systems which use the Golden code and shift register as PNGs have technology and methodology to synchronize their PNGs. As a next step to this development, we propose to implement methods and/or strategies which would be useful to synchronize cellular automata PNGs in the transmitter and in the receiver of spread-spectrum systems.

References

[1] J.G. Proakis, ¨Digital Communications¨, New York, Mc. Graw Hill, USA, 2001, pp. 726-791.

[2] R.L. Pickholts et. al., ¨Theory of the Spread Spectrum Communications, A Tutorial¨, IEEE Trans. on Com., vol. 30, no. 5, pp. 855-884, 1982.

[3] R. Muraoka et al, ¨Design and Implementation of a CDMA Transmitter for Mobile Cellular Communications¨, JART, vol. 1, no. 2, 127-136, 2003.

[4] Md. Abdul Alim, ¨Spread Sprectrum Modem for Voice and Data Transmission¨, JAIT, vol. 3, no. 2, pp. 115-119, 2012.

[5] S. Kalita and P. P Sahu, ¨A New Modified Sequence Generator for Direct Sequence Spread Spectrum (DSSS)¨, Nat. Conf. of Electr., Comm. and Sig. Proc., 2011, 144-146.

[6] K.Thenmozhi, Padmariya Praveenkumar et.al., ¨OFDM+CDMA+Stego=Secure Communication¨, RJIT, vol. 4, no. 2, pp. 31-46, 2012.

[7] R. Nuñez, ¨An Optimal Chaotic Bidirectional Communicator for Hidden Informatioin, Based on Sychronized Lorenz Circuits¨, JART, vol. 2, no.1, pp. 5-20, 2004.

[8] C. Posadas Castillo et. al., ¨Experimental Realization of Binary Signals Transmission Based on Synchronized Lorenz Circuits¨, JART, vol. 2, no. 2, pp. 127-137, 2004.

[9] G. Heidari-Betani and C.D. McGillem, ¨Chaotic sequences for spread spectrum: an alternative to PN-sequence¨, IEEE Inter. Conf. on Selected Topics in Wireless Commun, Vancouver, Canada, 1992, 437-440.

[10] D.F. Drake and D.B. Williams, ¨Pseudo-chaos for direct-sequence spread-spectrum communication¨, Proc. SPIE: Chaotic Circuits for Communication, vol. 2612, 1995, 104-114.

[11] H. Wang Hi and Hu Jiandong, ¨Chaotic Spread-Spectrum Communication Using Discrete-Time Synchronization¨, J. of China Univ. of Posts and Telecom., vol. 4, no. 1, pp. 66-69, 1997.

[12] D. Leon et. al., ¨Pseudo-chaotic PN-sequence generator circuits for spread spectrum communications, Circuits¨, Dev. and Sys., IEEE Proc.., vol. 151, no. 6, 2004, pp. 543.

[13] Xia Yongxiang et. al., ¨Correlation Properties of Binary Spatiotemporal Chaotic Sequences and Their Application to Multiple Access Communication¨, Phys. Rev. E, vol. pp. 64, 067201, 2001.

[14] L. De Micco et. al., ¨Zipping characterization of chaotic sequences used in spread spectrum communication systems¨, AIP Conf. Proc., XV Conference on Nonequilibrium Statistical Mechanics and Nonlinear Physics, vol. 913, 2007, 139-144.

[15] S. Wolfram, ¨A New Kind of Science¨, Champaign, Illinois, Wolfram Media Inc., 2002.

[16] S. Wolfram, ¨Random Sequence Generation by Cellular Automata¨, Ad. in Appl. Math., vol. 7, no. 2, pp. 123-169, 2004.

[17] S.Wolfram, ¨Cellular Automata and Complexity: Collected Papers¨, Boulder, Westview Pres, 1994.

[18] A. Rukhin et al., ¨A Statistical Test Suite for the Validation of Random Number Generators and Pseudo Random Number Generator for Cryptographic Applications¨, NIST Special Publication 800-22, 2001.