ISSN 0976 – 1411

Available online at www.internationaleJournals.com

INTERNATIONAL eJOURNAL OF MATHEMATICS AND ENGINEERING

www.InternationaleJournals.com

International eJournals

International eJournal of Mathematics and Engineering
2 (2016) Vol. 7, Issue 1, pp 7 – 16

Fibonacci Representations of Linear Feedback Shift Registers and its application for spread-spectrum system

Hossein Naraghi[1], M.M Motamedi-nezhad[2]and Hassan Naraghi[3]
[1]Department of Mathematics, PayameNoor University, P. O. Box: 19395-3697, Tehran, Iran.
[2]Department of Mathematics, University of Applied Science and Technology, Tehran, Iran
[3] Department of Electrical Engineering, Ashtian Branch, Islamic Azad University, Ashtian, Iran.
[1]Email: ho.naraghi@pnu.ac.ir , [2]Email: motamedi@uast.ac.ir ,
[3]Email: naraghi.hassan@yahoo.com

ABSTRACT

A Linear Feedback Shift Registers (LFSR) with "Fibonacci" architecture is a shift register provided with a small amount of memory which is used in the feedback algorithm [l]. Like linear feedback shift registers (LFSRs), FCSRs provide a simple and predictable method for the fast generation of pseudorandom sequences with good statistical properties and large periods. In this paper, we analyze an alternative architecture for LFSRs with "Fibonacci" architecture. We use Fibonacci sequences for BPSK and we determine n-state Markov chain entropy rate.

Keywords: BPSK, Direct-sequence,Entropy, Fibonacci mode, LFSR, Markov chain, Spread-spectrum.

1. INTRODUCTION

In the years since the publication of Golomb's book [6], the basic design of shift registers has been enhanced in several different directions. The "Galois" and "Fibonacci" modes were developed [8], many ways of interconnecting shift registers were analyzed, and perhaps most significantly, the binary state vacuum tubes were eventually replaced by cells with many possible states. Engineers were led, for example, to consider N-ary shift registers, whose cell contents are taken from the integers modulo N, or from a finite Galois field, or more generally from an algebraic ring. It turns out that much of the analysis of shift register sequences goes through in this more general setting. It is possible to enhance the basic shift register architecture in yet another way, by the addition of a small amount of memory. The memory is used as a "carry" in the calculations. For example, when two sequences of ones and zeroes are added, they can be

added as elements of $\mathbb{Z}/(2)$ (or XOR addition) in which 1+1=0, or they can be added as "integers", in which 1+1 = 2 = 0 + a carry of 1. The difference is illustrated by the following example, where carries go to the right:

1 1 0 0 0 1 1 0 1 0	1 1 0 0 0 1 1 0 1 0
1 0 1 0 1 1 0 1 1 0	1 0 1 0 1 1 0 1 1 0
0 1 1 0 1 0 1 1 0 0	0 0 0 1 1 0 0 0 1 1
mod 2	with carry

In fact, the summation combiner [5] does exactly the latter: it combines two binary sequences into a third, using addition with carry. It was originally proposed as a method for creating a difficult-to-predict bit stream from two relatively easy-to-predict bit streams, for cryptographic applications. Now, we give the definitions and describe the basic properties of linear feedback shift registers and linearly recurrent sequences. Throughout this chapter we assume that R is a commutative ring (with identity denoted by 1).

Definition 1.1. A (Fibonacci mode) linear feedback shift register of lengthm over$\{0,1\}$, with coefficients $q_1, q_2, ..., q_m \in \{0,1\}$ is a sequence generator (cf. Definition 2.1.2) whose state is an element
$s = (a_0, a_1, ..., a_{m-1}) \in \{0,1\}^m = \Sigma$, whose output is $output(s) = a_0$, and whose state change operation τ is given by

$$(a_0, a_1, ..., a_{m-1}) \to (a_0, a_1, ..., a_{m-1}, \sum_{i=1}^{m} q_i a_{m-i}).$$

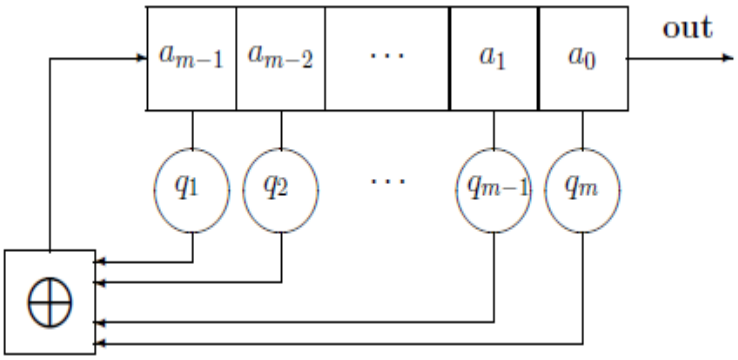

Figure 1. A Linear Feedback Shift Register of Length m.

(See also Section 3.1) It is convenient to think of a linear feedback shift register (or LFSR) as aphysical circuit, as pictured in Figure 1.Note that, because we like to think of bits as owing out to the right, the order of the components$a_i$in the diagram is the reverse of the order when we write the state as a vector. Whenthinking of LFSRs as physical devices as in the figure, we sometimes list the components of thestate in descending order of their indices. We write

a_{m-1}	a_{m-2}	….	a_1	a_0

for the machine stateto distinguish the two notations.A LFSR defines a sequence generator (R^m, R, τ, out) in the sense of Section 2.1.c. As usual fora sequence generator, from any given initial state (or "initial loading") $(a_0, \cdots, a_{m-1})$, the LFSR generates an infinite output sequence $a = a_0, a_1, \cdots, a_{m-1}, a_m, \cdots$

This sequence may also be described as a linearly recurrent sequence (see Section 2.2.b).

Definition 1.2. A sequence $a = a_0, a_1, \ldots$.of elements of R is linearly recurrent if there existsa finite collection $q_1, \ldots, q_m$ of elements of R such that for all $n \geq m$we have

$$a_n = q_1 a_{n-1} + \cdots + q_m a_{n-m}.$$

This equation is called the recurrence relation. The integer m is called the degree of the recurrence. The elements $q_1, \ldots, q_m$are called the coefficients of the recurrence, to which we mayassociate the connection polynomial

$$q(x) = \sum_{i=0}^{m} q_i x^i \in R[x].$$

A sequence $a = a_0, a_1, \ldots$.satisfies a linear recurrence with coefficients $q_1, \ldots, q_m$ if and only if it may be realized as the output sequence of aLFSR with connection polynomial $q(x) = \sum_{i=0}^{m} q_i x^i \in R[x]$. In particular, LFSR sequences are eventually periodic.

The phrase "the LFSR with connection polynomial q(x)"is ambiguous because we can add initial cells (with no feedback taps) to an LFSR to obtain a new LFSR with the same connection polynomial q(x) but of length m >deg(q). In this case the output sequence will consist of an initial "transient"$a_0, \ldots, a_{m-d-1}$ followed by the periodic part of the sequence. The LFSR will output strictly periodic sequences if and only if its length m equals the degree of its connection polynomial, that is, if $q_m \neq 0$. In order to be explicit, we will sometimes refer to "the LFSR with connection polynomial q(x) whose length is the degree of q(x)".

Every periodic sequence a (of elements in a ring R) can be realized as the output of a LFSR (with entries in R): just take a LFSR whose length is a single period of a, and feed the last cell back to the first cell. The number of cells in the shortest LFSR that can generate a is called the linear complexity or equivalent linear span of a.

2. LFSR, FIBONACCI ARCHITECTURE

The purpose of this section is to recall some well-known results concerning LFSRs, in a way which will motivate our treatment of FCSRs. In the Fibonacci representation (see Fig. 1), the register is initially loaded with bits $a_0, a_1, \cdots, a_{m-1}$. The output sequence is given by the linear recurrence $a_t = \sum_{i=1}^{m} q_i a_{t-i}$ for $\geq m$. In this paper, we assume $q_i = 1$, for every $i = 1, \ldots, m$.

Here is a table of linear feedback shift registers (LFSR) with "Fibonacci" architecture.

Initial condition	Output	Output sequence
0 0	0	0 0 0
0 1	1	0 1 1
1 0	1	1 0 1
1 1	0	1 1 0

Table 1. Output sequence by $a_n = a_{n-1} + a_{n-2}\ mod\ 2$.

Initial condition	Output	Output sequence
0 0 0	0	0 0 0 0
0 0 1	1	0 0 1 1
0 1 0	1	0 1 0 1
1 0 0	1	1 0 0 1
1 1 0	0	1 1 0 0
1 0 1	0	1 0 1 0
0 1 1	0	0 1 1 0
1 1 1	1	1 1 1 1

Table 2. Output sequence by$a_n = a_{n-1} + a_{n-2} + a_{n-3}\ mod\ 2$.

Initial condition	Output	Output sequence
0 0 0 0	0	0 0 0 0 0
0 0 1 0	1	0 0 1 0 1
0 1 0 0	1	0 1 0 01
1 0 0 0	1	1 0 0 0 1
0 0 0 1	1	0 0 0 1 1
1 10 0	0	1 10 0 0
0 1 1 0	0	0 1 1 0 0
1 01 0	0	1 0 1 0 0
0 1 0 1	0	0 1 0 1 0
0 1 1 1	1	0 1 1 1 1
1 0 1 1	1	10 1 1 1
1 1 0 1	1	1 1 0 1 1
1 1 1 0	1	1 1 1 0 1
1 0 0 1	0	1 0 0 1 0
0 0 1 1	0	0 0 1 1 0
1 1 1 1	0	1 1 1 1 0

Table 3. Output sequence by$a_n = a_{n-1} + a_{n-2} + a_{n-3} + a_{n-4}\ mod\ 2$.

We use the Fibonacci codes for the spreading signal. For more information study direct-sequence (DS) spread spectrum and binary phase-shift keying(BPSK) direct-sequence spread spectrum[4]. The simplest from of DS spread spectrum employs binary phase-shift keying(BPSK) as the spreading modulation. Ideal BPSK modulation results in instantaneous phase changes of the carrier by 180 degree and can be mathematically represented as a multiplication of the carrier by function c(t) which takes on the values +1 and -1. Consider a constant-envelope data-modulated carrier having power P, radian frequency ω_0, and data phase modulation $\theta_d(t)$ defined by

$$s_d(t) = \sqrt{2P}\cos[\omega_0 t + \theta_d(t)].$$

This signal occupies a bandwidth typically between one-half and twice the data rate prior to DS spreading, depending on the details of the data modulation. BPSK spreading is accomplished by multipling $s_d(t)$ by a function c(t) representing the spreading waveform, as illustrated in Figure 2. The transmitted signal is

$$s_t(t) = \sqrt{2P}c(t)\cos[\omega_0 t + \theta_d(t)]$$

This signal is transmitted via a distortionless path having transmission delay T_d. The signal is received together with some type of interference and/or Gaussian noise. Demodulation is accomplished in part by remodulation with the spreading code appropriately delayed as shown in Figure 3. This remodulationor correlation of the received signal with the delayed spreading waveform is called despreading and is a critical function in all spread-spectrum system. The signal component of the output of the despreading mixer is

$$\sqrt{2P}c(t - T_d)c\left(t - \hat{T}_d\right)\cos[\omega_0 t + \theta_d(t - T_d) + \phi]$$

Where$\hat{T}_d$ is the receiver's best estimate of the transmission delay.

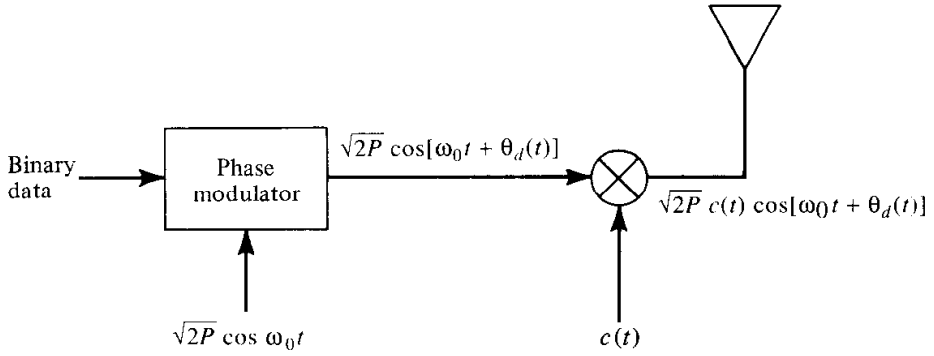

Figure 2. BPSK direct-sequence spread-spectrum transmitter.

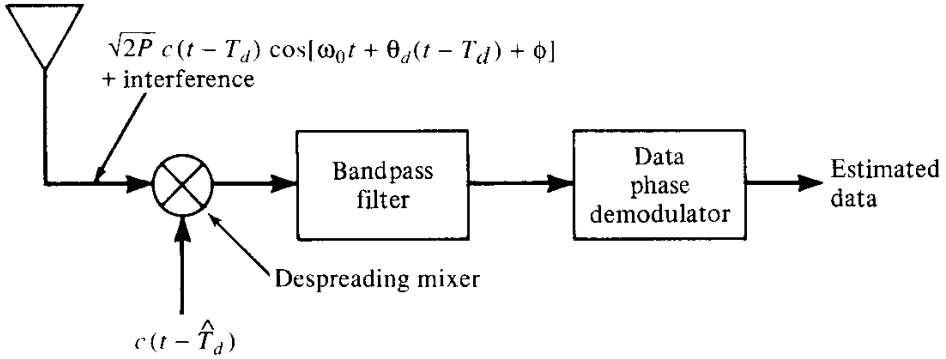

Figure 3. BPSK direct-sequence spread-spectrum receiver.

The two-side power spectral density in W/Hz of a binary phase-shift-keyed carrier is given by

$$S_d(f) = \frac{1}{2}PT\{sinc^2[(f - f_0)T] + sinc^2[(f + f_0)T]\}$$

which for more details see [4].

We compare the data and spreading waveform by random code and Fibonacci code with an initial shift-register load of 00001 are illustrated in Figure 4.a for random code and Figure 4.b for Fibonacci code

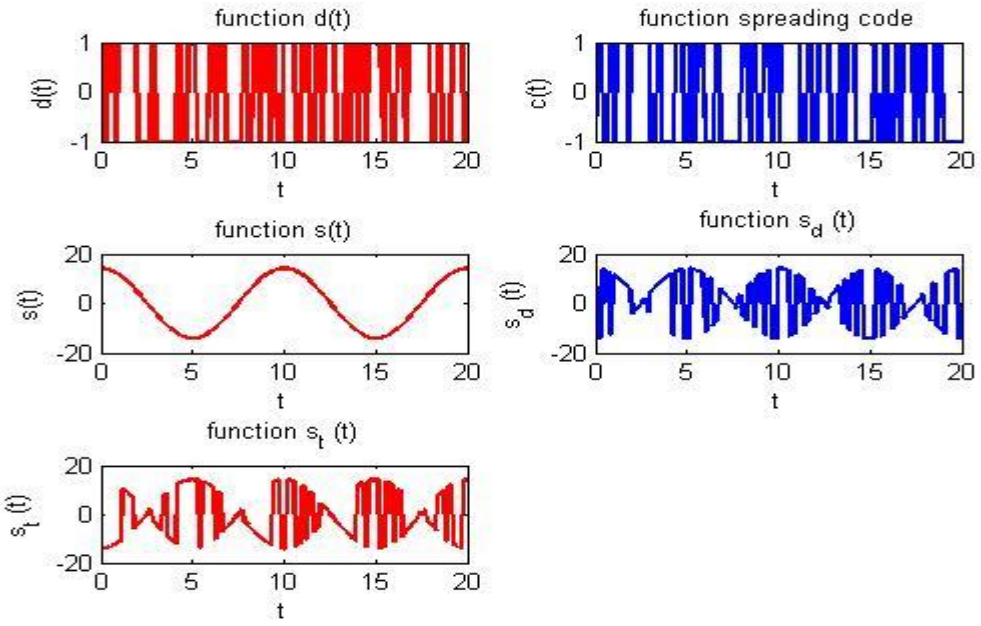

Figure 4.a

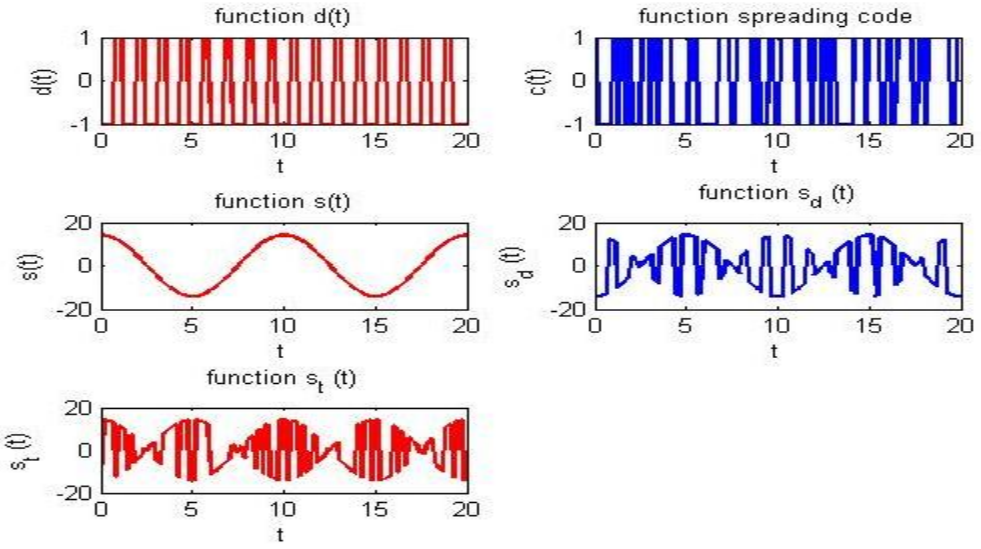

Figure 4.b

The two-side power spectral density in W/Hz of a Fibonacci code with an initial shift-register load of 00001 are illustrated in Figure 5.

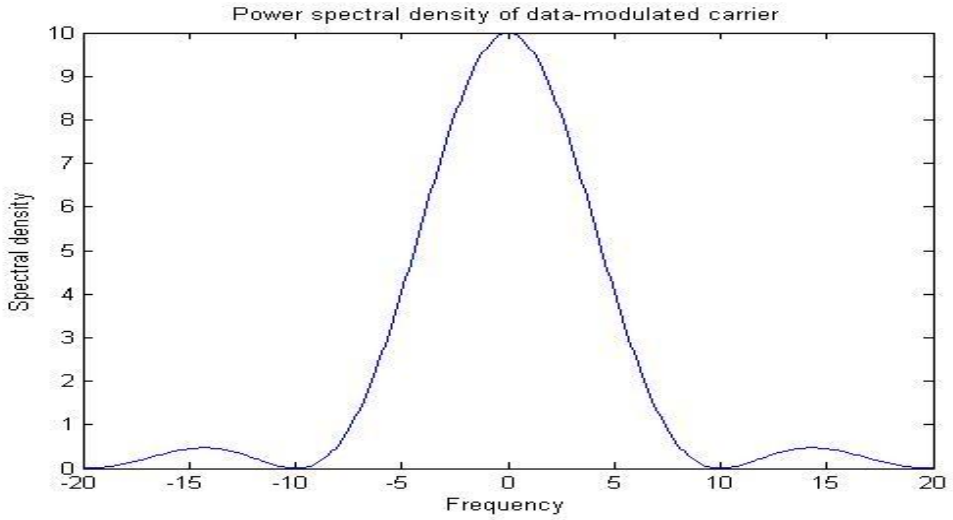

Figure 5

3. Entropy rate

In this section, whenever possible we follow the notation and terminology of [7].
The entropy of a random variable X with a probability mass functionp(x) is defined by

$$H(X) = -\sum_{x} p(x) \log p(x).$$

We use logarithms to base 2. The entropy will then be measured in bits. The entropy is a measure of the average uncertainty in the random variable.It is the number of bits on average required to describe the random variable.

The joint entropy H(X,Y) of a pair of discrete random variables (X, Y) with a joint distribution p(x, y) is defined as

$$H(X,Y) = -\sum_{x\in X}\sum_{y\in Y} p(x,y) \log p(x,y)\,.$$

If (X, Y) ~p(x, y), the conditional entropy H(Y|X) is defined as

$$H(Y|X) = \sum_{x\in X} p(x)\, H(Y|X=x) = -\sum_{x\in X}\sum_{y\in Y} p(x,y) \log p(y|x)\,.$$

If we have a sequence of n random variables, a natural question to ask is: How does the entropy of the sequence grow with n? We define the entropy rate as this rate of growth as follows.

We will assume that the Markov chain is time invariant unless otherwise stated.

If $\{X_i\}$is a Markov chain, $X_n$is called the state at time n. A time invariant Markov chain is characterized by its initial state and a probability transition matrix$P = [P_{ij}]$, i, j ∈ {1, 2, . . . , m}, where$P_{ij} = \Pr\{X_{n+1} = j|X_n = i\}$.

If it is possible to go with positive probability from any state of the Markov chain to any other state in a finite number of steps, the Markov chain is said to be irreducible. If the largest common factor of the lengths of different paths from a state to itself is 1, the Markov chain is said to aperiodic.

Definition 3.1. The entropy of a stochastic process $\{X_i\}$is defined by

$$H(\chi) = \lim_{n\to\infty} \frac{1}{n} H(X_1, X_2, \dots, X_n)$$

when the limit exists.For a stationary Markov chain, the entropy rate is given by

$$H(\chi) = \lim_{n\to\infty} H(X_n|X_{n-1}, \dots, X_1) = \lim_{n\to\infty} H(X_n|X_{n-1}) = H(X_2|X_1),$$

where the conditional entropy is calculated using the given stationary distribution. Recall that the stationary distribution μ is the solution of the equations

$$\mu_i = \sum_i \mu_i P_{ij} \text{ for all j.}$$

We express the conditional entropy explicitly in the following theorem.

Theorem 3.2. Let $\{X_i\}$be a stationary Markov chain with stationary distribution η and transition matrix P. Let$X_i \sim \mu$. Then the entropy rate is

$$H(\chi) = -\sum_{ij} \mu_i P_{ij} \log P_{ij}.$$

Proof. $H(\chi) = H(X_2|X_1) = \sum_i \mu_i (\sum_j -P_{ij} \log P_{ij})$.

Consider a n-state Markov chain with a probability transition matrix$P = [P_{ij}]$.Let the stationary distribution be represented by a vector μ whose components are the stationary probabilities of states 1,2,…,n, respectively. Then the stationary probability can be found by solving the equation $\mu P = \mu$or, more simply, by balancing probabilities. For the stationary distribution, the net probability flow across any cut set in the state transition graph is zero.
If the Markov chain has an initial state drawn according to the stationary distribution, the resulting process will be stationary. The entropy of the state $X_n$at time nis

$$H(X_n) = H(\mu_1, \mu_2, \dots, \mu_n).$$

However, this is not the rate at which entropy grows for$H(X_1, X_2, \dots, X_n)$. The dependence among the *Xi*'s will take a steady toll.

Example 3.3. Consider a two-state Markov chain with a probability transition matrix

$$P = \begin{bmatrix} 1-\alpha & \alpha \\ \beta & 1-\beta \end{bmatrix}$$

as shown in Figure 6.Weobtain$\mu_1\alpha = \mu_2\beta$. Since $\mu_1 + \mu_2 = 1$, the stationary distribution is

$$\mu_1 = \frac{\beta}{\alpha+\beta}, \mu_1 = \frac{\alpha}{\alpha+\beta}.$$

The entropy of the state X_n at time n is

$$H(X_n) = H\left(\frac{\beta}{\alpha+\beta}, \frac{\alpha}{\alpha+\beta}\right) = H(X_2|X_1) = \mu_1 H(\alpha) + \mu_2 H(\beta) = \frac{\beta H(\alpha) + \alpha H(\beta)}{\alpha+\beta}.$$

The entropy rate is at most 1 bit because the process has only two states. This rate can be achieved if (and only if) $= \beta = {}^1\!/_2$, in which case the process is actually i.i.d. with $\Pr(X_i = 0) = \Pr X_i = 1) = {}^1\!/_2$.

Consider a two-state Markov chain with a probability transition matrix

$$P = \begin{bmatrix} 1-p & p \\ 1 & 0 \end{bmatrix}.$$

As a special case of the general two-state Markov chain, the entropy rate is

$$H(X_2|X_1) = \mu_1 H(p) + \mu_2 H(1) = \frac{H(p)}{p+1}.$$

By straightforward calculus, we find that the maximum value of H(X) which occurs for $p = \frac{3-\sqrt{5}}{2} = 0.382$.The maximum value is $H(p) = H(1-p) = H\left(\frac{\sqrt{5}-1}{2}\right) = 0.604\ bits$.
Note that $\frac{\sqrt{5}-1}{2} = 0.618$ is (the reciprocal of) the Golden Ratio. See Figure 7.

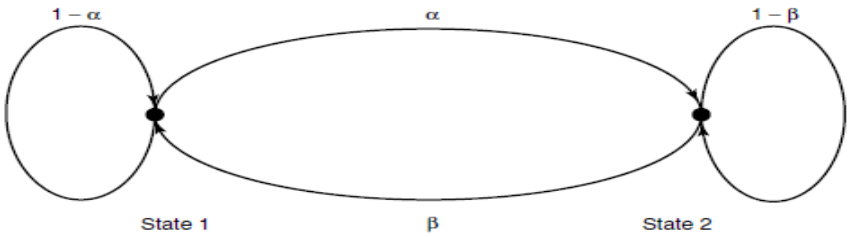

Figure 6. Two-state Markov chain

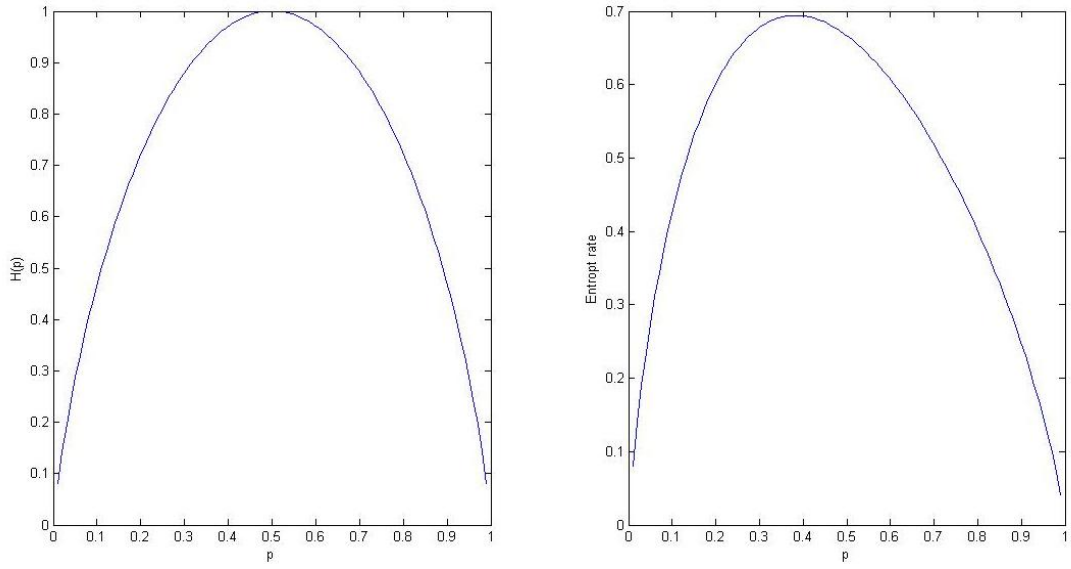

Figure 7.

Let N(t) be the number of allowable n- state sequences of length t for the Markov chain such that $t > n$. Consider any allowable sequence of symbols of length t . If the first symbol is defined, then the remaining N(t-1) and if the first and second symbol are defined, then the remaining N(t-2) and if symbols are defined until the n-th symbol; the remaining $N(t-n)$ symbols can form any allowable sequence.So the number of allowable sequences of length t satisfies the recurrence

$$N(t) = N(t-1) + N(t-2) + \cdots + N(t-n).$$

For t=2 and n=2, we could choose N(1)=2, and N(2)=3 (in this case the sequence 11 is not allowed). The sequence N(t) grows exponentially, that is, $N(t) \approx c\lambda^t$, where λ is the maximum magnitude solution of the characteristic equation$1 = z^{-1} + z^{-2}$. Solving the characteristic equation yields $\lambda = \frac{1+\sqrt{5}}{2}$,the Golden Ratio. Suppose that $H_0 = \lim_{t\to\infty} \frac{1}{t}\log N(t)$ .Therefore$H_0 = \lim_{t\to\infty} \frac{1}{t}\log N(t) = \log\left(\frac{1+\sqrt{5}}{2}\right) = 0.694$ bits.

REFERENCE

1. A. Klapper and M. Goresky, Feedback shift registers, 2-adic span, and combiners with memory, *J.* Crypt. 10 (1997), 111-147.

2. J. T. Barrows, Jr., A new method for constructing multiple error correcting linear residue codes, Rep. R-277, Coordinated Sci. Lab., Univ. of Illinois, Urbana, 1966. 360.

3. R. E. Blahut, Theory and Practice of Error Control Codes, Addison-Wesley, Reading MA, 1983. 402, 507.

4. R. L. Peterson, R. E. Ziemer and D. E. Borth, "Introduction to spread spectrum communications", Prentice-Hall Inc., 1995.
5. R. Rueppel, New approaches to stream ciphers, Ph.D. thesis, Swiss Federal Institute of Technology, Z□urich, 1984. 16, 411.

6. S. Golomb, Shift Register Sequences. Holden-Day, San Francisco, CA, 1967. 15, 213, 273, 312, 501.

7. T. M. Cover and J. A. Thomas, "Elements of information theory", John Wiley and Sons, Inc., 1991.

8. W. W. Peterson and E. J. Weldon, Jr., Error-Correcting Codes second edition, MIT Press, Cambridge MA, 1972. 15, 168, 201.