Enhanced LSB technique for Audio Steganography Using Random sample Encoding

Ashraf A. Hosny, Wael A. Murtada, Mohamed I. Youssef

Abstract— Least Significant Bit (LSB) hiding technique is the most simple and efficient technique used for audio steganography. The idea behind this research is to improve the conventional LSB technique by increasing the capacity of the hidden data and to get minimum effect for audio signal, which is used to hide data. The approach proposes two steps; the first step is to randomize the selected frame that is used to hide data while the second step to embed the message bits in the deeper layers of samples and alter other bits to decrease the error, to achieve higher capacity and robustness.

Index Terms— Audio signal, Bit modification, Data Hiding, Least Significant Bit, Randomization, Robustness, Steganography.

1 Introduction

Today the Steganography deals with many electronic media rather than physical objects the Medias are used for digitally embedding message such as plain text hyper text audio or video still images and network traffic. There are many Steganography techniques which are used for hiding the data in a cover file. Broadband communication networks and multimedia data available in a digital format opened many challenges and opportunities for innovation. Multimedia information hiding is widely used to protect personal privacy and many effective methods have made progresses over the recently years. The word Steganography comes from the Greek and it means covered or secret writing [2]. Nowadays, embedding techniques are used for hiding information into something else for the sole purpose of hiding that information from the casual spectator.

Audio Steganography describe methods to embed information into a carrier signal. In this technique audio file is sampled and then an appropriate bit of each alternate sample is altered to embed the textual information [4] [8].

2 Features Of Steganography

Embedding of data into an audio signal should be with the following restrictions and features:

1. The data should be in encrypted format and directly embedded into the media, instead of embedding into a header or cover, so that the data remain undamaged across varying data file formats [1][3].
2. The audio signal should not be corrupted and the embedded data should not be traceable. Data must be hidden, inaudible, not recognizable, and unseen to mean that an observer does not notice the presence of the data, even if they are perceptible[1] [3].
3. The embedded data should be immune to modifications variety from planned and intellectual wound at removal to anticipated manipulations.
4. Asymmetrical coding of the embedded data is desirable, since the purpose of data hiding is to keep the data in the host signal, but not necessarily to make the data difficult to access.
5. Error detection and correction coding should be used to ensure data integrity [1] [3] [4].

Characteristics of Steganography:

- Confidentiality
- Imperceptibility
- Accurateness
- High capacity
- Resistance
- Visibility
- Survivability
- No detection

The main idea of Steganography is to provide secure data at the receiver end like the cryptography. Both have been used to protect information. The cryptographic technique scramble messages so if intercepted, the messages cannot be understood. The Steganography involves making the content of the secret message unreadable while not preventing non intended observers from learning about its existence[9][10]. The goal of Steganography is to hide the data from third party whereas the goal of cryptography is to make data unreadable by third party.

3 Steganography Techniques

Echo Hiding:

Encodes and echoes the secret message in the form of the binary forms in audio signal with minimal degradation at the data. In echo hiding information is embedded in a sound file by introducing an echo into the discrete signal, Echo Hiding places embedded message in cover audio by introducing an echo.

- *Ashraf A. Hosny, Researcher at National Authority for Remote Sensing and Space Sciences (NARSS), Cairo, Egypt. ashraf.adel@narss.sci.eg*
- *Wael A. Murtada, Ph.D., On-Board Computer and Space Software Department, National Authority for Remote Sensing and Space Sciences (NARSS), Cairo, Egypt. Wael_murtada@narss.sci.eg*
- *Mohamed I. Youssef, Prof., Electronics and Communications Department, Al-Azhar University, Cairo, Egypt. drmiyoussef@yahoo.com*

LSB Technique:

It is the way to embed information in a digital audio file. By substituting the least significant bit of each sampling point with a binary message, LSB coding allows for a large amount of data to be encoded.

Parity Coding:

It is way to break the sound signal into areas then hides the message in the parity bit. If the parity does not match, it adjusts the LSB of one of the samples to get the required (even) parity.

Spread Spectrum:

In the context of audio Steganography, the basic spread spectrum (SS) method attempts to spread secret information across the audio signal's frequency spectrum as much as possible [6]. Spread spectrum makes use of the fact that small changes are more difficult for the human eye or ear to detect at high energy levels (loud audio or bright video). The message is hidden in those areas of the carrier file with the greatest energy.

4 Proposed Algorithm

In this research, we improve the conventional LSB algorithm, by embedding the message bits to the audio bit stream (16 bit per sample) in random samples (section 5.0) to increase robustness, then apply bit modification technique (section 6.0) on the samples to decrease the error and to maintain the high perceptual transparency of the resulting audio signal

In our proposed model we take two consecutive bits from the secret message and instead of changing a single bit in a sample we change two bits (for example 3rd and 4th position) of the sample. If there is change in these two bits we flip rest of the LSB otherwise there is no change. For example, if the original sample value was $(0...01000)_2=(8)_{10}$, and the hidden message bits “01” are to be embedded into 3rd and 4th LSB layer, the standard algorithm will produce the value $(0...00000)_2=(0)_{10}$ to embed the 1st bit only and for the 2nd bit we need another sample, to embed the 2nd bit. Our proposed algorithm will produce $(0...00111)_2 = (7)_{10}$ to embed the two bits in one sample and after the sample reshaping, this sample is much closer to the original sample and contains two hidden bits (here 0 and 1) instead of one bit.

Steps of data Encoding algorithm:

1. Get the Tn input text to be embedded. Where n=1...m. Length of secret text is m, where m must be less than 65535 characters.
2. Convert the text into ASCII binary code.
3. Read the An input audio file as cover file. Where n=1...k. The total number of samples of audio cover is k.
4. Check the condition k > (m *8) +16. If yes, then execute the embedding process. Where 16 is the number of bits represent the total size of the message required to be hidden.
5. Embed the size of the message (m represented in bits) in the first 16 samples in the LSB of the WAV file.
6. Select the sample according to the PN sequence generator
7. In the audio sample (k samples) hide the binary codes of secret text in the corresponding LSB bits of WAV file.
8. Sample reshaping to get the minimum effect due to the message hiding.
9. Repeat above procedure from point 6, till the entire message embedded in audio.

Steps of data Decoding algorithm:

1. Read the cover audio file
2. Extract the size of the hidden message by reading the LSB of the first 16 samples
3. Extract the binary code by reading the samples of LSB according to the PN sequence generator
4. Convert binary code into characters.
5. Display the secret message Tn.

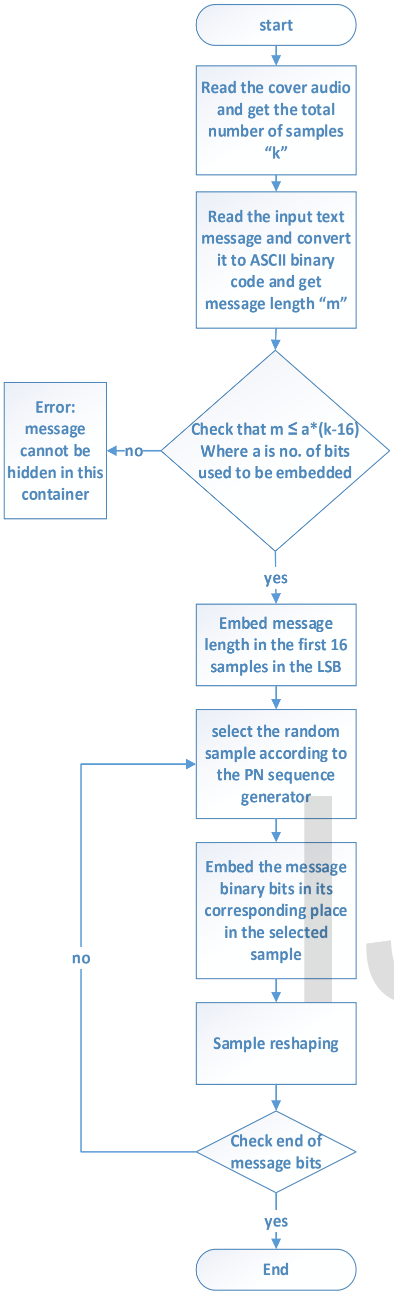
start
Read the cover audio and get the total number of samples "k"
Read the input text message and convert it to ASCII binary code and get message length "m"
Check that m ≤ a*(k-16) Where a is no. of bits used to be embedded
no
Error: message cannot be hidden in this container
yes
Embed message length in the first 16 samples in the LSB
select the random sample according to the PN sequence generator
Embed the message binary bits in its corresponding place in the selected sample
Sample reshaping
Check end of message bits
no
yes
End

Fig.1: Encoding algorithm

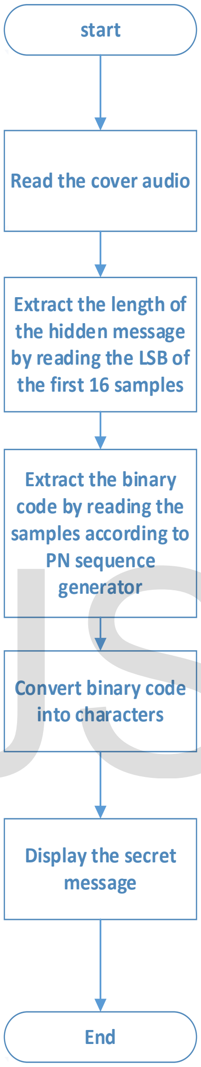
start
Read the cover audio
Extract the length of the hidden message by reading the LSB of the first 16 samples
Extract the binary code by reading the samples according to PN sequence generator
Convert binary code into characters
Display the secret message
End

Fig.2: Decoding algorithm

5 Sample Selection using PN Sequence Generator

Another way to confuse the intruder is to add some more randomness in secret message embedding by using PN sequence random generation for the selected sample to hide the secret message.

5.1 PN Codes

The linear feedback shift registers (LFSR) generators are shift registers with feedback connections that use xor-gates

5.2 Linear feedback shift registers (LFSR)

The design of the PN code uses the so-called "generating polynomial", which is the representation of the feedback connections in the form of polynomial, f(D), where D is the unit delay in the LFSR. The feedback connections for these LFSR generators are of either Galois configuration or Fibonacci configuration. Galois configuration uses summing units (XOR) between the shift-register stages, whereas Fibonacci configuration uses them in the feedback path of the shift register.

Linear-feedback shift registers with n-stages may produce a sequence of the maximum possible length of length 2n – 1. These sequences are called maximum LFSR sequences or m-sequences ↾ [13], ↾ [14], ↾ [16], ↾ [17], ↾ [18], and their generating polynomials (f(D)), are called primitive polynomials. The number of one's in an m-sequence is larger than the number of zero's by one, and the sequence contains at most (n-1) successive zeros.

5.3 Maximization of Sequence Period for a Fixed Register Length

The maximum period achievable by a shift register with n memory elements is 2^n. This maximum period is reduced by 1 if the generator uses linear feedback, because the zero state is self-perpetuating, i.e., The LFSR initiated with zeros produces the uninteresting all-zeros sequence. Thus, the most efficient LFSR generators must cycle through all possible non-zero states before repeating, and the period L of an LFSR's state sequence is bounded by:

$$L \leq 2^n - 1 \qquad (1)$$

Certain generating polynomials produce sequences of maximum length of 2^n – 1. These sequences are called m-sequences and their generating polynomials are called primitive polynomials.

In our algorithm we use this technique to generate random sequence for the sample selection that is used to hide the secret message into it.

This means that the embedded secret bits will not be arranged in order but it will be controlled by using the shift register sequence generator shown fig.3 for the sample selection.

The polynomial that used is known at both the transmitter and the receiver side. It consists of 17 shift registers, for maximum length the polynomial used is $x^{17} + x^{14} +1$, which gives 131,071 sequences and that is sufficient for this application, but if you need to hide larger capacity of data, it may need to change to a larger cover audio and increase the number of sequences.

Next it is required to set the unit delays by an initial state values, the output random sequence will be changed by changing these values, so it is important that the initial values must be fixed and known at both transmitter and receiver also this gives more security to the hidden message.

Finally convert the output bits to integer value then add it to 16, as the first 16 samples is reserved for embedding the size of the hidden message to the LSB of each sample as shown in fig. 3.

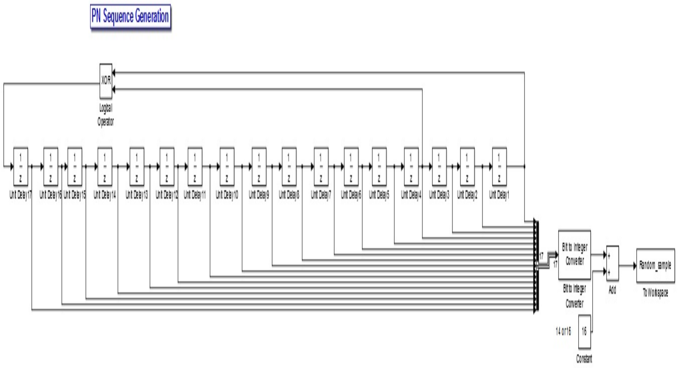

Fig. 3 Random PN sequence generator

6 Sample Reshaping

This method is able to shift the limit for transparent data hiding in audio from lower LSB layer to higher LSB layer, using a two- step approach. In the first step, the secret bit is embedded into the ith LSB layer of the host audio using a LSB coding method. In the second step, the other bits in the sample will be modified in order to decrease the noise caused by this embedding. The standard LSB coding method simply replaces the original host audio bit in the ith layer (i=1,...,16) with the bit from the secret message bit stream. In the case when the original and secret message bit are different.

The ith LSB layer is used for embedding the error caused by the hiding process is 2^i [1] quantization steps (QS) with amplitude range is [-32768, 32767] (this for hiding single bit). The embedding error is positive if the original bit was 0 and watermark bit is 1 and vice versa. The key idea of the proposed LSB algorithm is to minimize the distortion caused by the hiding process and at the same time not loosing the information of the hidden message in samples. It is clear that, if only one of 16 bits in a sample is fixed and equal to the secret message bit, the other bits can be flipped in order to minimize the embedding error and to be much closer to the original one. However, the extraction algorithm remains the same; it simply retrieves the ith bit by reading the bit value from the predefined LSB layer in the audio sample.

7 Test Results

To evaluate the performance of the proposed audio steganography algorithm, two wav audios were used.

As defined in [12], signal to noise ratio for the WAV signal in time domain is computed using the relation:

$$SNR = 10.\log_{10} \frac{\sum_n x^2(n)}{\sum_n [x(n)-y(n)]^2} \qquad (2)$$

Where x(n) represent sample of input audio sequence and y(n) stands for sample of audio with modified LSBs.

SNR is calculated for both audios using the same hidden message; the message is hidden in all channels;

Maximum capacity of the message can be hidden in audio when hiding 1 bit per sample is calculated by:

[(Total number of samples * number of channels) – 16] / 8 (3)

Table 1 & 3 show the SNR when using the maximum capacity for both cover audios.

Table 2 shows the SNR of the 2nd audio when using the same message size of the 1st audio.

Wav (A) properties:
Number of Channels: 2
Sample Rate: 48000
Total number of Samples: 42672
Duration: 0.889 sec
Bits per Sample: 16

The maximum size of hidden text used for WAV (A) is 21332 characters except in hiding in the 3rd LSB only; it was half the size of the message (10666 characters).

Table 1 WAV (A) SNR at the maximum capacity using all samples for hiding the text message in different layers

Hidden message placement inside the sample	SNR (dB)
3rd & 4th LSB with standard method	53.80
3rd LSB with standard method	62
2nd & 3rd LSB with standard method	59.82
3rd & 4th LSB with proposed algorithm	61.14
2nd & 3rd LSB with proposed algorithm	66.12

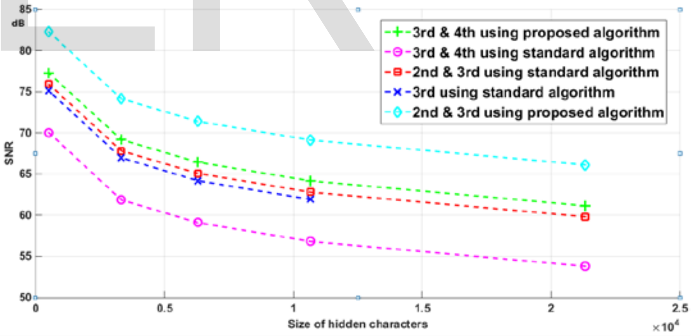

Fig.4 WAV (A): SNR vs different size of hidden message characters

Wav (B) properties:
Number of Channels: 2
Sample Rate: 44100
Total number of Samples: 48558
Duration: 1.1011 sec
Bits per Sample: 16

Table 2 WAV (B) SNR at the same hidden message of WAV (A) in different layers

Hidden message placement inside the sample	SNR (dB)

3rd & 4th LSB with standard method	56
3rd LSB with standard method	63.74
2nd & 3rd LSB with standard method	62
3rd & 4th LSB with proposed algorithm	62.23
2nd & 3rd LSB with proposed algorithm	67.63

The maximum size of hidden text used for WAV (B) is 24275 characters; SNR is calculated as shown in table 3 with maximum capacity while hiding using the 3rd LSB only the size was 12137 characters.

Table 3 WAV (B) SNR at the maximum capacity using all samples for hiding the text message in different layers

Hidden message placement inside the sample	**SNR (dB)**
3rd & 4th LSB with standard method	55.45
3rd LSB with standard method	63.17
2nd & 3rd LSB with standard method	61.48
3rd & 4th LSB with proposed algorithm	61.67
2nd & 3rd LSB with proposed algorithm	66.80

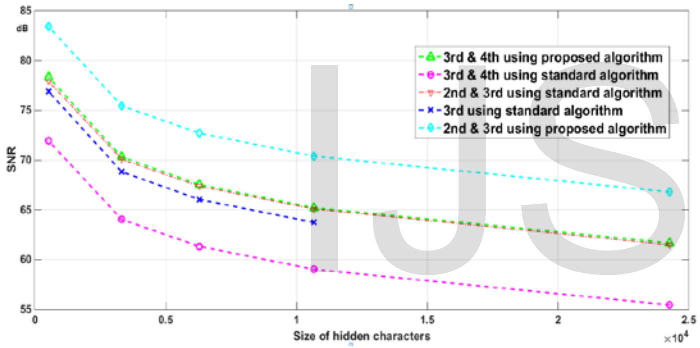

Fig.5 WAV (B): SNR vs different size of hidden message characters

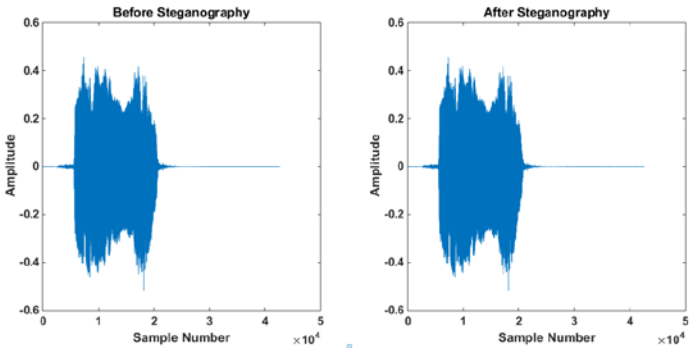

Fig. 6 WAV (A) before and after audio stegnography at maximum capacity

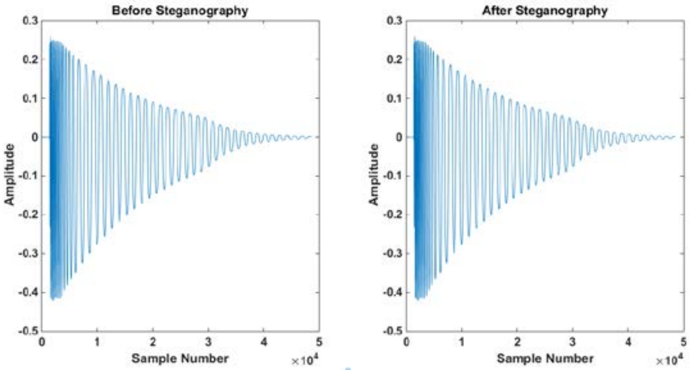

Fig. 7 WAV (B) before and after audio stegnography at maximum capacity

It is clear from the previous tables and figures that the proposed algorithm improves the SNR over the standard method in comparing with the same order of bits used, also the results in SNR while using the for example the 3rd and 4th bits in proposed algorithm *(WAV A: 61.14 dB – WAV B: 61.67 dB*) is better than using 2nd & 3rd LSB standard algorithm *(WAV A: 59.82 dB – WAV B: 61.48 dB)*, this gives more robustness as we go to deeper layers without affecting the SNR.

In addition it is very close to the 3rd bit only *(WAV A: 62 dB – WAV B: 63.17 dB)*, this means that the proposed algorithm succeeded to give the same results in spite of using a higher bit and using 2 bits to hide the message data in the proposed algorithm gives more capacity rather than hidding only one bit, all of that is done while maintaining the transparency with the same cover (fig 6 & fig. 7).

7.1 Advantages of the proposed approach

- Described algorithm succeeds in not only increasing the depth of the embedding layer but also sample was chosen randomly without affecting the perceptual transparency of the audio signal.
- That is, two-way of robustness, first, the selected frame is randomly chosen, Second, Additive noise has less effect on the hidden message as a higher order bits are modified.
- The proposed algorithm obtains significantly lower bit error rate than the standard algorithm.
- The stega-analysis of the proposed algorithm is more challenging as well, because there is a significant number of bits flipped in a number in bit layers and the opponent cannot identify exactly, which bit layer is used for the data hiding.
- In addition, tests showed that the algorithm succeeds in increasing capacity, while keeping SNR value close to the level of SNR obtained by standard LSB embedding with lower capacity.

8 Conclusion

This research paper has extended the conventional LSB modification technique for audio steganography to make it more secure against steganography analysis. An intelligent algorithm used to embed the message bits in the deeper layers of samples and alter other bits to decrease the error. Objective tests showed the algorithm succeeds in increasing capacity, while keeping SNR value close to the level of SNR obtained by standard LSB embedding with lower capacity. The stego message formed on the basis of proposed methodology cannot be differentiated from host message. The secret message on the receiver side can be extracted from the stego message as well.

REFERENCES

[1] Anuradha, Kriti, Harish "Audio Steganography step toward the secure data transmission: An overview" National Conference on emerging computing technology ISBN Number 978-81-89547-85-1 , 2010.

[2] Nedeljko Cvejic ,Tapio Seppben " Increasing the capacity of LSB Based audio Steganography", IEEE 2002.

[3] S.K.Moon, R.S.Kawitkar "Data Security using Data Hiding" International Conference on Intelligence and multimedia Application. IEEE 2007

[4] Tanmay Bhowmik, Pramatha Nath Basu "On Embedding of text in Audio- A case of Stegnography" International conference on recent trends in information, Telecommunication and computing, IEEE 2010.

[5] H. B.Kekre, Archana Athawale, Uttara Athawale "Information Hiding in Audio Signals" International Journal of Computer Applications (0975 - 8887) Volume 7- No.9, October 2010

[6] R Sridevi, Dr A Damodaram, Dr SVL.Narasimham , "Efficient Methods Of Audio Steganography By ModifiedLSB And Strong Encription" Journal of Theoretical and Applied Infonnation Technology 2009

[7] Matsuoka, H ,"Spread spectrum audio Steganography using sub-band phase shifting", Proceeding of the 2006 International conference on intelligent infonnation hiding and multimedia signal processing, IEEE2006

[8] Pradeep Kumar Singh, Hitesh Singh,Kriti Saroha " A Survey on Steganography in Audio" National Conference on Computing for Nation Development, Indiacom 2009

[9] R.Anderson,F.petitcolas On the Limits of Steganography, IEEE journal selected areas in communication, Vo1.16,No 4, 1998.

[10] Ping Wah Wong and E J Depl editor Security and watermarking of Multimedia contents vohne 3657.Society of Photo-optical Instrumentation Engineers 1990

[11] K. Gopalan "Audio Steganography using bit modification ", proc.lEEE Int. conf acoustics, speech, and signal processing Vo1 2, pp 421-424. April 2003.

[12] P. Bassia, I. Pitas, N. Nikolaidis, “Robust audio watermarking in the time domain,” IEEE Transactions on Multimedia, vol3, 2, June 2001.

[13] R. L. Peterson et al., “Introduction to Spread-Spectrum Communications”, Prentice Hall, 1995.

[14] M. K. Simon and J. K. Omura, “Spread Spectrum Communications Handbook”, McGraw-Hill, 2002.

[15] D. Torrieri, “Principles of Spread-Spectrum Communication Systems”, Springer, 2005.

[16] J. J. Spilker, “Digital Communications by Satellites”, Prentice-Hall, 1977.

[17] R. L. Pickholtz, “Theory of spread spectrum communications – a tutorial,” IEEE transaction Comm., Vol. 30, pp. 855 – 884, May 1982.

[18] S. W. Golomb, “Shift Register Sequences”, Aegean Park Press, 1982.