The effectiveness of symbolic integer splitting method over both synchronous stream ciphers and perfectly secret ciphers

A H Alhussain

Department of Information Technology, Peoples' Friendship University of Russia, Moscow, Russia.

E-mail: amanie.alhussain@mail.ru

Abstract. Synchronous stream ciphers are classified up to the used keystream into two types: either to the synchronous ciphers that are used pseudorandom keystreams, or to the perfectly-secret ciphers that are used truly random keystreams; each of these classifications has its disadvantages, so the necessity to find a method that belongs to the synchronous stream ciphers and overcomes their disadvantages is required. In this article, the author proposes a method to satisfy this purpose; the proposed method is called symbolic integer splitting method and it is a mathematical method for representing an integer in form of a certain sequence of integers by using the modular arithmetic operation. Also this method can be classified under perfectly-secret ciphers because it satisfies asymptotic secrecy under a certain condition. The new scientific results are conducted as the ability of this method to success in dealing with the disadvantages of both the traditional synchronous stream ciphers and perfectly-secret ones, likes: insecure protection of information, hide the information about the length of original message from the cryptanalyst, no need to use each gamma only once, not expensive in terms of required resources to store gammas.

Keywords: Vernam cipher, synchronous stream cipher, Symbolic splitting method, Absolute secrecy, asymptotic secrecy, Perfectly-secret ciphers.

1. Introduction

A synchronous stream cipher is a stream cipher, in which the keystream is generated independently of ciphertext or plaintext [1, 2]. The block diagram of this cipher is shown in figure 1.

![Figure 1. The block diagram of the synchronous stream cipher.](image)

A keystream is a stream of random or pseudorandom characters. The "characters" in the keystream can be bits, bytes, numbers or actual characters like A-Z depending on the usage case. In this article, the keystream will be called as gammas. Two methods are used to generate the keystream sequences: the first method is a random sequence: used for Vernam cipher and one time pad; while the second one is a deterministic sequence: also known as pseudorandom sequence, which is generated by using pseudorandom number generators (PRNG) and are used in both gamma cipher and binary additive stream cipher. Studies show that the synchronous stream ciphers face problems according to the used
keystream. Here are some examples of the problems (disadvantages) in two different ciphers which are classified under the synchronous stream ciphers.

The first example is Vernam cipher, which is considered as a synchronous stream cipher that uses a truly random keystream sequence. Vernam cipher is a type of symmetric cipher, which was invented by AT&T employee Gilbert Vernam in 1918. Vernam cipher method can be described by the following equations [3, 4]:

\[ Y = X \oplus K = (x_1 \oplus k_1, x_2 \oplus k_2, \ldots, x_l \oplus k_l) \text{ - Vernam encryption process} \]  
\[ X = Y \oplus K = (y_1 \oplus k_1, y_2 \oplus k_2, \ldots, y_l \oplus k_l) \text{ - Vernam decryption process,} \]

where, \( X = x_1, x_2, \ldots, x_l \) is the binary source message (plaintext) of length \( l \), \( K = k_1, k_2, \ldots, k_l \) is the binary key string of the same length \( l \), \( Y = y_1, y_2, \ldots, y_l \) is the ciphertext, and \( \oplus \) represents the bitwise operation "exclusive or".

Studies have shown that Vernam cipher has serious key management problems, like: key generation and key distribution or and the need to deliver the same keystream to the receiver side as the sender side, also the need to safely store the keystream for both the sender and receiver, all of these problems lead to the fact that Vernam cipher is not practical for general use [5].

The second example is gamma cipher, which is classified under synchronous stream cipher that uses a pseudorandom keystream sequence. The ciphertext in gamma cipher is generated by applying the logical exclusive-or operation between the plaintext and the keystream (sequence of gammas), which is generated by using a pseudorandom number generator (PRNG) [6]. Gamma cipher can be executed using several mathematical formulas. For example, the encryption process can be performed by the following formula [7]:

\[ C = P \oplus K, \]  

where \( C, P, K \) – ASCII codes of the ciphertext, plaintext and gamma, respectively, \( \oplus \) – bitwise operation - "exclusive or". The decryption process (plaintext restoration) is performed similarly by using the following formula [7]:

\[ P = C \oplus K. \]  

Studies have shown that gamma cipher solves the problems that are found in the Vernam cipher related to the key generation and key distribution by using PRNG which requires as input just a small amount of information called a secret key, but at the same time a new problem arises because PRNG has a period, or by other words there is a predictability of the generated pseudorandom numbers (gammas), since there is a relationship between them [8], and as a result this problem leads to the insecure protection of information by using gamma cipher.

The problems that are discussed earlier guide us to put the first objective of the current study which can be summarized as the necessity to create a method that overcomes the problems of both key managements, which is included key generations and key distributions, and insecure protection of information.

Both Vernam cipher and one time pad belong to other class of the ciphers which is called perfectly-secret cipher. According to Shannon, perfectly-secret ciphers mean that the ciphertext \( C \) and the plaintext \( M \) are statistically independent, i.e. for all possible values of \( M \) and \( C \) satisfy the following probability [9, 10]:

\[ \Pr(M / C) = \Pr(M) . \]  

The above formula allows us to conclude that the ciphertext \( C \) does not give the cryptanalyst any information about the plaintext \( M \).

Studies have shown that perfectly-secret ciphers have the following problems: they provide cryptanalyst information about the length of the original message (plaintext) [11], each gamma is used only once, so the gammas are very expensive in terms of the required resources, also there is a problem of the demand to store gammas [12]. All of the mentioned problems conduct us to state the second objective of the current study which can be summarized as the need to find a cipher that
satisfies secrecy, hides information about the length of the plaintext, and does not require using each gamma only one time, i.e. not expensive in terms of the required resources.

All of the above discussed problems will be solved by a method that is called symbolic integer splitting method. This method is proposed by the author [13-14]. It is important to notice that in spite of the different publications by the author about this method but the effectiveness and importance of the proposed method over the above mentioned ciphers are not published in any publications and this is the main objective of the current study to show how integer splitting method solves their disadvantages and problems.

2. Materials and methods
The applied method is called symbolic integer splitting method and it is proposed by the user [13-14] and can be defined as follows:

The condition of applying the integer splitting method is described by the following equation:

\[ r > a > 0, \]

where \( r \) – represents the keystream (sequence of gammas), \( a \) – the numerical value of the plaintext character in the selected code table.

2.1. The proposed integer splitting method.
The splitting method is explained by the following definition proposed by the author [13-14]:

**Definition 1.** The integer splitting of the number \( a \) on the basis of \( r \) is the representation of \( a \) as a sequence of numbers \( a_1, a_2, a_3, \ldots, a_{k-1}, a_k \) in which

\[
a_1 = \delta^{(2)}, \text{ where } \delta^{(2)} = r \mod a,
\]

\[
a_2 = \delta^{(3)}, \text{ where } \delta^{(3)} = r \mod q^{(2)}, \quad q^{(2)} = \left\lfloor \frac{r}{a} \right\rfloor,
\]

\[
a_3 = \delta^{(4)}, \text{ where } \delta^{(4)} = r \mod q^{(3)}, \quad q^{(3)} = \left\lfloor \frac{r}{q^{(2)}} \right\rfloor,
\]

\[......\]

\[
a_{k-1} = \delta^{(k)}, \text{ where } \delta^{(k)} = r \mod q^{(k-1)}, \quad q^{(k-1)} = \left\lfloor \frac{r}{q^{(k-2)}} \right\rfloor,
\]

\[
a_k = q^{(k)}, \text{ where } q^{(k)} = \left\lfloor \frac{r}{q^{(k-1)}} \right\rfloor,
\]

where \( \delta \) – is the remainder of the integer division \( r/a \), and \( q \) – is the integer part of this division, and the symbol \( \left\lfloor \right\rfloor \) means rounding down to the nearest integer. The natural number \( k \) is called the level of splitting.

2.2. The proposed technical method to generate the base (r), which represents the gamma.
For generation the base \( r \), which is represented the keystream, the author proposed a technical method [15] to improve the randomness level of PRNG by the usage of what is called deterministic genetic algorithm (DGA), the details about how to perform it and its results is available in the author’s publication [15]. I will display just two tests to be used during the conduction of the comparisons with gamma ciphers (in the mentioned publication five tests are available).

To analysis the benefits of the proposed deterministic genetic algorithm (DGA), the following steps are followed:

- Build two programs using MATLAB: the first program for generation the keystream sequence by using traditional PRNG, while the second program for generation the keystream sequence by using PRNG with the proposed DGA.
- Collect the obtained data from both programs and analysis them by using the statistical programs (Minitab and IBM SPSS Statistics), after that, conduct the results.

Table 1 show the results of two tests for three pseudorandom number generators.
**Table 1.** The statistical tests to verify the effectiveness of using deterministic genetic algorithm with traditional three pseudorandom number generators when generating the keystream \( r \).

| Test                      | Linear congruential generator | Blum BlumShuber generator | Quadratic congruential generator |
|---------------------------|-------------------------------|---------------------------|----------------------------------|
| Without DGA              | With DGA                      | Without DGA               | With DGA                         | Without DGA | With DGA |
| Wald–Wolfowitz runs test  | 0                             | 0.048                     | 0                                | 0.361       | 0       |
| Entropy as a measure of randomness | 1.58                          | 6.37                      | 1.14                             | 5.30        | 1       |
|                           |                               |                           |                                  |             | 5.11    |

The statistical significance level for all tests equals to \( a = 0.01 \). The null and alternative hypothesis for Wald–Wolfowitz runs test are: \( H_0 \) – The distribution in the sequence is random; \( H_1 \) – the distribution in the sequence is not purely random.

From Table 1 we conclude the following results about Wald–Wolfowitz runs test: all the p-values of the generated gammas for the three traditional generators without applying any DGA satisfy the following inequality: \((p\text{-value } = 0.00) < (a = 0.010)\), so the null hypothesis \( H_0 \) about the randomness distribution in the sequences is rejected and the alternative hypothesis \( H_1 \) about not purely random distribution in the sequences is not rejected. On the contrary, all the p-values of the generated gammas for the three traditional generators with the usage of DGA satisfy the following inequality: \( p\text{-value } > a \), then the null hypothesis \( H_0 \) about the randomness distribution in the sequences is not rejected.

Entropy is a measure of disorder or unpredictability of the information elements. The higher it is, the more chaotic, unpredictable. From Table 1 can be seen that, (the entropy of traditional PRNG without using DGA) < (the entropy of the traditional PRNG using DGA), so the sequences of gammas which are obtained from traditional PRNG using a DGA are more chaotic (unpredictable) compared to the sequences generated without the usage of DGA.

2.3. The asymptotic secrecy of the integer splitting method.

The integer splitting method satisfies an asymptotic secrecy as it has been proven by the author in the publication [14], this property is satisfied *in the condition of* increasing the splitting level \( k \), this lead to providing the following definition of the asymptotic secrecy:

**Definition 2.** We say that a method that depends on a parameter \( k \) has asymptotic secrecy if it verifies

\[
\text{when } k \to \infty \Rightarrow \Pr(M \mid C, k) \to 0
\]

where, \( \Pr(M \mid C, k) \) – represents the probability of restoring the plaintext \( M \) from the sequence \( C \) (ciphertext) that occurs as a result of splitting method at the splitting level \( k \).

2.4. The steps of performing the proposed method.

The steps to perform the simple integer splitting method are summarized as follows:

1) The secret key should be available at sender and receiver sides, which is consists of the parameters: PRNG, DGA and the splitting level \( k \).
2) A keystream should be generated on the base of the secret key mentioned at the step 1 (table 1 shows some of the statistical tests results of this keystream).
3) Convert each character in the plaintext into its numerical value in the selected code table.
4) Select the value of gamma in the keystream that is generated at step 2 that satisfies the condition of applying integer splitting method, i.e. \( r \) (gamma) > maximum value of the selected code.
5) Use the integer splitting method that is presented in definition 1 to encrypt the plaintext character by using numerical value that is obtained at step 3 with the usage of the base (gamma) that is obtained at the step 4.

6) Save the result of the splitting method for each plaintext character.

7) Repeat the steps 3-6 until reach the end of the plaintext.

2.5. The ability to reuse the gamma more than one time.

Lemma 1. The integer splitting method solves the condition of the necessity to use each gamma only once, which is found in some synchronous stream ciphers like Vernam cipher.

Proof.

Let $c_1$ and $c_2$ – the ciphertexts, which are obtained when encrypting the messages $m_1$ and $m_2$, accordingly by the usage of exclusive-or operation $\oplus$ with the same gamma $r$:

\[
\begin{align*}
    c_1 &= m_1 \oplus r \\
    c_2 &= m_2 \oplus r
\end{align*}
\]

(8)

(9)

Exclusive-ORing the two ciphers together $\Rightarrow$ is independent of gamma, because exclusive-or something with itself results in zero.

\[
c_1 \oplus c_2 = m_1 \oplus r \oplus m_2 \oplus r = m_1 \oplus m_2
\]

(10)

In this case, the text can be restored based on the existence of probabilistic words and probabilistic languages in the texts (standard beginning or ending, parts of n-grams, etc.). This method of decryption is called "pulling a probable word."

If the splitting method, which is described in definition 1 is used: the result of encryption $m_1$ on the base of $r$ (gamma), i.e. ciphertext $c_1$, will be consisted of three components $a_1, a_2, a_3$ if the splitting level equals to 3, and the result of encryption $m_2$ on the base $r$ (gamma), i.e. ciphertext $c_2$, will be consisted of three components $b_1, b_2, b_3$ if the splitting level equals to 3.

Exclusive-ORing the results of these two ciphers together $\Rightarrow a_1 \oplus a_2 \oplus a_3 \oplus b_1 \oplus b_2 \oplus b_3$

(11)

None of the values $a_1, a_2, a_3, b_1, b_2, b_3$ or not all of them represent the values of ASCII code of a single character, because of the method that is applied to obtain these different components of the ciphertexts $c_1$ and $c_2$ which are calculated either by the usage of different values of modules in case of $a_1, a_2, a_3, b_1, b_2, b_3$, or by applying the division operations in case of $a_1$ and $b_1$, so the plaintexts cannot be extracted in a way that is based on the existence of probability words in texts or probabilistic language.

3. Results

The integer splitting method can be considered to be classified under the class of synchronous stream cipher because the keystream (sequence of gammas) is generated independently of the ciphertext and the plaintext. Remember that the gamma represents $r$ in the definition of the integer splitting method. The generation of $r$ depends on PRNG parameters and DGA parameters as discussed in previous section and in publication [13], so a comparison is made in form of table 2 between the proposed method and some traditional synchronous stream ciphers to show the effectiveness of the proposed method.
Table 2. Comparison between two types of synchronous stream ciphers and symbolic integer splitting method.

|                                   | Vernam cipher | Gamma cipher | Symbolic integer splitting cipher |
|-----------------------------------|---------------|--------------|-----------------------------------|
| The period of Gammas (keystream period) | Absent, One-time pad | Available [8] | Significantly increased by the usage of deterministic genetic algorithm with pseudorandom number generators [15] |
| The need to store gammas          | Yes           | No           | No                                |
| The need to deliver the same sequence of gammas (keystream) to the receiver and sender | Yes           | No           | No                                |
| Secure protection of information  | Absolute secrecy | No           | Asymptotic secrecy [14]            |

The problems of the perfect secrecy ciphers are solved by the proposed integer splitting method as shown in the comparison presented in table 3.

Table 3. Comparison between perfectly-secret ciphers and symbolic integer splitting cipher.

|                                   | Perfectly-secret ciphers | Symbolic integer splitting cipher |
|-----------------------------------|--------------------------|-----------------------------------|
| Necessity to use each gamma only once | Yes [12]                | No                                |
| Provide cryptanalyst information about the length of the original message (plaintext) | The information about the length of the original plaintext is shown | The information about the length of the original plaintext is hidden |
| The need to store gammas          | Yes [12]                | No                                |
| Absolute secrecy                  | Yes                      | No, but provide asymptotic secrecy [14] |

4. Discussion

Table 2 presents the benefits of using symbolic integer splitting method in comparison with traditional synchronous stream ciphers (Vernam cipher that uses a truly random keystream and gamma cipher that uses a pseudorandom keystream), and shows that the proposed method is an optimal solution to solve the different problems that face the two synchronous stream ciphers because of its various properties that can be summarized as follows:

– The period of the sequence of gammas (keystream) is quite large according to the fact that symbolic splitting method with the help of both deterministic genetic algorithm parameters and pseudorandom number generators will create a sequence of gammas that are more random according to several well-known criteria, as was shown in Table 1 and in the publication [15]. This property of the symbolic splitting method solves a serious problem arising during the implementation of the gamma cipher [2–4], which is associated with the need to generate a long period sequence of gammas.

– The symbolic splitting cipher solves a serious problem that arises when implementing the Vernam method, which is associated with the need to deliver the same gammas’ sequence to the receiver as the sender, or the need to safely storage identical key sequence (gammas) for the sender and receiver. This problem has been solved by symbolic splitting cipher by using a secret secret key that generates the same sequence of gammas (keystream) at the sender side and the receiver one. Therefore, instead of transferring or storing large sequence of gammas as in the Vernam cipher, it is just enough to transfer (or save) a small amount of information called secret
key, its components are described in details in the publication [13] and in section materials and methods.

– The symbolic integer splitting cipher satisfies asymptotic secrecy as described in details in the publication [14], so it overcomes the disadvantage that is found in the gamma cipher, which is associated with the insecure protection of information.

So the first objective of the current research, which is connected to the necessity to create a synchronous stream method that has the features of both key managements (key generation and key distribution) and secure protection of information, is satisfied.

Table 3 demonstrates that there are two advantages of symbolic integer splitting cipher in comparison with traditional perfectly-secret ciphers:

– The first advantage: The splitting cipher is practical and not very expensive in terms of required resources compared to perfectly-secret ciphers, since it does not require to fulfil the condition of using each gamma only once, as discussed in lemma 1.

– The second advantage: the symbolic splitting cipher hides the information about the length of the original message (plaintext) from the intruder. The length of the ciphertext, which is encrypted using symbolic splitting method, is determined as follows: \( c = k \times l \), where \( c \) – is the length of the ciphertext, \( k \) – is the level of splitting, and \( l \) – is the length of the plaintext, while \( k \) – is unknown.

So the second objective of the current research is satisfied, and it is related to the requirement to create a cipher that has the features of hiding information about the length of the original message from the cryptanalyst (second mentioned advantage), not requiring to use each gamma only once (lemma 1), not demanding to store or transferring a large sequence of gammas (instead it uses a small secret key that will generate the same sequence at the receiver side and sender one) also providing asymptotic secrecy [14].

5. Conclusions

In conclusion, we can sum up that the symbolic integer splitting method solves some problems that are found in both synchronous-stream ciphers and perfectly-secret ciphers. So it is proposed as an optimal solution that merges the advantages of both ciphers and overcomes most of their disadvantages. What makes it special is its ability to change its level of protection of information by changing the level of splitting. The future research will be conducted to show the effectiveness of this method in comparison with the substitution ciphers, which are based on the modular arithmetic operation, and Chinese remainder theorem because the decryption process of the splitting cipher meets the goal of this theorem.

6. References

[1] Niederreiter H 2002 Coding Theory And Cryptology (World Scientific) p 460
[2] Goutam P and Subhamoy M 2011 RC4 Stream Cipher and Its Variants (CRC Press) p 311
[3] Wenbo M 2005 Modern cryptography: theory and practice (Moscow: Williams) p 768
[4] Charles P P and Shari L P 2003 Security in Computing (Prentice Hall Professional) p 746
[5] Keith M 2017 Everyday Cryptography: Fundamental Principles and Applications (Oxford University Press) p 674
[6] Alferov A P, Zubov A Yu, Kuzmin A S and Cheryomushkin A B 2005 Fundamentals of cryptography (Moscow: Helios ARV) p 480.
http://enigma.grsu.by/index.php?option=com_content&view=article&id=20004:2012-03-08-19-50-29-20004&catid=41:2012-02-18-18-00-13&Itemid=69
[7] Agranovsky AV and Khadi RA 2009 Practical cryptography: algorithms and their programming (Moscow: Solon-Press) p 256.
https://www.studentlibrary.ru/book/ISBN5980030026.html
[8] Michael J 2000 Proceedings of the Military, Government and Aerospace Simulation Symposium (Society for Computer Simulation International) p 269

[9] Sebastian M and Catuscia P 2012 Theory of Security and Applications: Joint Workshop (Springer Science & Business Media) p 225

[10] Hans D and Helmut K 2015 Introduction to Cryptography: Principles and Applications (Springer) p 508

[11] Hans D and Helmut K 2002 Introduction to Cryptography: Principles and Applications (Springer Science & Business Media) p 310

[12] Johannes B 2004 Introduction to Cryptography (Springer Science & Business Media) p 338

[13] Stefanyuk V L and Alhussain A H 2016 Symmetric encryption on the base of splitting method Bulletin of PFUR Series Mathematics. Information Sciences. Physics № 2 p 53–61

[14] Alhussain A H 2020 Asymptotic secrecy of the information protection by the usage of simple integer splitting method IOP Conference Series Materials Science and Engineering 862 052032

[15] Alhussain A H and Stefanuk V L 2015 Using Genetic Algorithm to improve periodic level of pseudorandom number generators «The First European Conference on Informational Technology and Computer Science»: East West Vienna Austria p 25–34