Cryptanalysis of Alternating Step Generators

Robert Wicik       Rafał Gliwa       Piotr Komorowski
Cryptology Department
Military Communication Institute
Zegrze, Poland
{r.wicik, r.gliwa, p.komorowski}@wil.waw.pl

Abstract—Alternate clocking of linear feedback shift registers is the popular technique used to increase the linear complexity of binary sequences produced by keystream generators designed for stream ciphers. The analysis of the best known attacks on the alternating step generator led us to add nonlinear filtering functions and the nonlinear scrambler to the construction. In this paper we give complexities of these attacks applied to the modified alternating step generator with nonlinear filters and the scrambler. We also suggest minimum lengths of registers in the original alternating step generator to make it resistant to the attacks.

Keywords—stream cipher; keystream generator; feedback shift register

I. INTRODUCTION

Stream ciphers are symmetric key ciphers used for information protection in telecommunication channels. Such type of cipher is often suitable for hardware implementation which may provide high level of security. A keystream generator is the basic component of most stream ciphers. It is a pseudorandom generator of bit or digit sequence which is used for encryption of plaintext. The same generator produces the sequence for decryption of ciphertext. Generators implemented at a sender and a receiver side should produce the same keystream in synchronous manner.

Linear feedback shift registers (LFSR) are basic components of keystream generators in classical stream ciphers. An LFSR of length \( m \) with properly selected feedback function gives sequence of maximal period \( 2^m - 1 \) and good statistical properties but has low linear complexity – only \( m \). It is vulnerable to Berlekamp-Massey [2] algorithm and can be easily reconstructed having short output segment of length just \( 2m \). Stop-and-go or alternate clocking of shift registers are two of methods to increase the linear complexity of the keystream. Other techniques introduce nonlinear functions to define feedbacks or filters of shift registers. All these methods increase the linear complexity as well as resistance of keystream generators to reconstruction of their internal states.

The alternating step generator (ASG) [4] is an example of the keystream generator, where the de Bruijn sequence [1] controls irregular clocking of two linear feedback shift registers. Despite the high linear complexity, the ASG is vulnerable to various attacks [5..14], so there are many modifications of this generator [15..18]. Interesting are ASG\( (r,s) \) [5,6,7] and modified alternating \( k \)-generators (MAG\( k \)) [8]. In [20] we proposed four more: MASG, MASG\( 0 \), MASG\( 1 \) and MASG\( 2 \) – the MASG family. From the analysis of the attacks on the ASG and its modifications we conclude that the linear function (XOR) at the output of such type of keystream generator should be replaced with nonlinear one.

In [22] we propose to add a nonlinear scrambler at the output of the alternating step generators: ASG, MAG\( k \) and MASG\( i \). We constructed the nonlinear scrambler with the nonlinear feedback shift register [20]. Particular realization of this idea is the MASG\( 15 \) keystream generator, with implemented nonlinear scrambler, nonlinear filtering functions and the initialization method.

In this paper, we describe selected attacks on the alternating step generators. Then we try to apply these attacks to the generator with nonlinear filters and the scrambler to assess their resistance to cryptanalysis. Also we estimate minimum lengths of registers in the original alternating step generator to make it resistant to the attacks.

II. ALTERNATING STEP GENERATORS

Alternating step generators are pseudorandom generators of binary keystream sequences, where the concept of stop-and-go [3] shift registers was adapted.

A. The original Alternating Step Generator

First alternating step generator (ASG) was proposed in [4]. The ASG consists of two linear feedback shift registers of lengths \( m_1, m_2 \), alternately clocked by \([1]\) the de Bruijn sequence of length \( k \). The de Bruijn sequence of period \( K=2^k \) can be easily obtained by adding zero bit after \( k-1 \) zeros in the sequence with period \( 2^k-1 \) taken from the LFSR. The exclusive-or sum (XOR) of bits from irregular clocked LFSRs produces output bits from the generator.

![Figure 1. The Alternating Step Generator](image)

\[ \text{clock} \quad \text{LFSR}_1 \quad 2^{m_1-1} \quad \text{LFSR}_2 \quad 2^{m_2-1} \quad \text{keystream} \]

\[ \text{de Bruijn seq.} \quad 2^k \]

\[ \text{electric} \]
For properly selected feedback polynomials, the output sequence from the ASG has large period $T$ (1) and high linear complexity $L$ (2):

\[ T = M_1 M_2 2^k \]
\[ (m_1 + m_2) 2^{k-1} < L \leq (m_1 + m_2) 2^k \]

where $M_1$ and $M_2$ are periods of LFSRs, and $LFSR_2$, respectively, and $M_1, M_2 > 1$, $\text{GCD}(M_1, M_2)=1$.

We can observe growth of the linear complexity of the output sequence from the ASG in comparison to the sequence obtained from a simple LFSR.

The ASG is vulnerable to various attacks. There are many variants of correlation and algebraic attacks and the best two are described in [9] and [14]. Asymptotic time complexity of these attacks is $O(m^2 2^{m/3})$ and data complexity is $O(2^{2m/3})$, where $m$ is the length of the shorter register from two alternately clocked. The time complexity of the algebraic attack described in [18] is much higher, however this attack can be applied if polynomials of irregular clocked registers are known, while requiring less output bits. These attacks exploit dependencies between output sequence (for known plaintext) and internal states of irregularly controlled registers.

B. Modified Alternating Step Generators

In order to resist the ASG against the attacks, a lot of modifications of this generator were proposed. In the alternating step $(r, s)$ generator [5], two integers $r$ and $s$ determine how many times registers LFSR$_r$ and LFSR$_s$ are clocked. In [16] authors showed, that the ASG($r,s$) is as secure as the original ASG. Afterwards, Kanso proposed in [7] and [8] MGCCASG and MCCASG constructions, where integers $r$ and $s$ are variable – dependent on a key or on a function of the controlling register state.

Another method of improving the ASG, proposed in [15], was to replace some of LFSRs with feedback with carry shift registers (FCSR) and XOR sum for addition with carry (ADD) as an output function. This modification of the ASG does not improve its security substantially.

Modified alternating $k$-generators (MAG$_k$) were proposed in [8]. Output sequence from MAG$_k$ is produced by the XOR sum of binary sequences from all three shift registers. Feedback functions of these registers can be linear or nonlinear.

1. MAG$_k^1$ – the function of state bits of the controlling register determines how many times controlled registers are clocked – this generator is similar to MCCASG [7];
2. MAG$_k^2$ – the binary output of the function (inner control function) of controlling register state bits determines alternating clocking of controlled registers – this generator was analyzed in [17], where authors showed that its security is not better than the security of the original ASG;
3. MAG$_k^3$ – the output from the generator is produced by the function (output generating function) of binary states of all three registers: one controlling and two controlled ones – we checked this concept in MASG$_2$ generator [20], but it gives non-random sequences.

C. The MASG family

In [19] we proposed a family of modified alternating step generators (MASG). We concentrated on selecting proper nonlinear functions – ones as feedback functions and other ones as filtering and combining functions of shift registers.

Our first approach to modification of the ASG was to replace controlled registers (LFSR$_r$ and LFSR$_s$) by nonlinear feedback shift registers (NLFSR). In [19] and [21] there are described methods for constructing nonlinear feedback functions for shift registers. At this time, we can achieve registers with maximal period for length up to $n=31$. These registers give sequences with the linear complexity close to the period, maximum $2^{n-2}$. MASGs with NLFSRs are:

- MASG – the ASG, where the output is produced by the XOR sum of binary sequences from two alternately clocked NLFSRs;
- MASG$^0$ – the ASG, where the output is produced by the XOR sum of binary sequences from all three registers – one LFSR and two NLFSRs (like in MAG$_k$).

These MASGs produce binary sequences with better linear complexity than the ASG, but we should find NLFSRs with greater length than 31 ($n=64$, in order to avoid brute force attacks on these registers). NLFSRs should give sequences with maximal period and high linear complexity.

Nonlinear Boolean functions are often used as filtering or combining functions for linear feedback shift registers in order to increase security of keystream generators. Functions proposed in [20] have high nonlinearity and many nonlinear components in their algebraic normal form. These functions we used in our second approach to modification of the ASG and we achieved:

- MASG$_1$ – the MAG$_k$, where all three LFSRs are equipped with nonlinear filtering functions;
- MASG$_2$ – the MAG$_k$ with three LFSRs and one nonlinear output function.

Output sequences from these constructions have better linear complexities than the ASG. MASG$_1$ gives sequences, which seem to be random, while MASG$_2$ does not pass randomness tests [20].
The most of attacks on the alternating step generators are divide-and-conquer attacks with known plaintext. Main goal is to find initial states of shift registers having a portion of the output sequence.

A. Divide-and-conquer attack

Divide-and-conquer attack was presented by C. G. Günther in [4], when describing original ASG. The basis of the attack is that the output sequence may be divided into two parts, derived from regularly clocked registers. Then these subsequences can be tested for a low linear complexity in an easy way using Berlekamp-Massey algorithm. If tested sequence with a period of $2^k$ is consistent with a sequence from clock control register, then the linear complexity of component sequences for irregularly clocked registers is lower than their periods. The time complexity of the divide-and-conquer attack, if one knows only feedbacks of the register, for which initial state is searched, is $O(m_1^k + m_2^k)$. When one knows feedbacks of all registers, then the complexity of the attack is $O(m_1^k m_2^k)$. If all registers are added to the set of known registers, then the complexity is $O(m_1^k m_2^k)$ and instead of linear complexity – linear consistency test is applied. In both cases, guessing clock control register is necessary.

B. Edit distance correlation attack

To carry out the edit distance correlation attack [10], it must be assumed, that feedbacks of irregularly clocked registers are known and the clocking sequence is irregularly characterized by a uniform distribution of bits 0 and 1. The attack involves searching the entire space of initial states of alternately clocked registers with known feedbacks, followed by verifying whether they are appropriate. Verification is based on the Hamming distance between the computed segment of the output sequence (obtained as the output of the generator with fixed states of alternately clocked registers) and the segment obtained as a result of the attack with a known plaintext. This distance is the minimum number of necessary subtractions (edit distance) in the computed segment, which allows obtaining known output sequence. Minimum is calculated for all $2^{2k}$ states of clock control register. There exists [10] effective method of calculating the distance and it is possible to determine the probability, that this distance is equal to 0, i.e. when initial sequences give known output sequence of the generator for the specified clock control sequence. This probability increases with the length of known segment of the output sequence.

The length of required known segment of the output sequence is linear in relation to the sum of lengths of irregularly clocked registers. The number of multiple solutions is minimized when the available output sequence is 4 times longer than total length of registers, which are searched. The time complexity of this attack is $O((m_1 + m_2)^2 2^{m_1 + m_2})$. The third register can be restored with the complexity $O(2^{0.274n})$, if only a sufficiently long segment of the sequence is available.

C. Edit probability correlation attack

The edit probability correlation attack on individual irregularly clocked registers in Günther generator was proposed in [11]. The attack uses probability (edit probability) that given segment of the output sequence of the generator has been produced from the sequence derived from regularly clocked register with predetermined initial state. Finding the initial state of one of the irregularly clocked registers can be done without knowledge of the other one and without knowledge about the state of clock control register.

The edit probability correlation attack requires a known output sequence with length minimum 4 times longer than length of state of the register, which is searched. The complexity of calculating this probability is the square of the length of output sequence. The time complexity of this attack, in order to find both initial states of the irregularly clocked registers is $O(\max^2 (m_1, m_2) 2^{\max(m_1, m_2)})$. For long registers, the complexity of edit probability correlation attack is much lower than the complexity of the edit distance correlation attack.

D. Reduced complexity correlation attack

In [9] it was proposed attack with reduced complexity on generators with irregularly clocked registers. In the output sequence of the generator, segment of consecutive zeros (or ones) is searched. It is assumed that half of them come from one of the irregularly clocked registers. This occurs with a certain probability. The remaining bits are obtained by exhaustive search. The optimal time complexity of this attack is $O(m_3^2 2^{3m_3})$ and requires $O(2^{3m_3})$ bits of sequence, where $m_3$ is the length of the register, which is searched: $m_1$ or $m_2$. These complexities apply to both the attack on the first and on the second irregularly clocked register.

In another scenario, the segment of some number of ones (or zeros) in the output sequence is searched and it is assumed that half of them come from one register, and the rest (ones and zeros) from the other. This occurs with a certain probability. The complexities of the attack according to this scenario are similar to these mentioned above for one register. Finding the initial state of the second register may require higher quantity of calculations.

E. New reduced complexity attack

New reduced complexity attack is based on a low resistance of Günther generator to sampling [14]. The low resistance to sampling indicates the possibility of effective finding all possible register’s preimages $A(Z^{\prime})$ of a generator, for a given segment of output sequence ($Z^{\prime}$). Generally, this resistance is defined as $2^n$, where $n$ is the maximum available length of output sequence.

In order to execute the attack, first, the set of all possible states for a given segment of output sequence of length $n$ is searched. Algorithm for finding this set is based on the divide-and-conquer attack with parity test. For all states of the initial clock control register, the output segment is divided into bits, originated from particular irregularly clocked registers. Then all states of irregularly clocked registers are checked, if they can generate separate bits – if so, the possible states of three registers are added to the set of $A(Z^{\prime})$, which is searched.

Average number of initial states of Günther generator for a given segment of output sequence is $2^{3m-n}$ for $n \leq 3m$, where $m$ is
length of registers, \( n \) – length of segment of output sequence. The time complexity of the algorithm is \( O(\max(2^n,2^{2m-n})) \). The complexity is determined by the factor \( 2^n \) when size of set of possible initial states is \( \leq 2^n \), otherwise it is determined by the value \( 2^{2m-n} \).

This algorithm can be effective and it shows low resistance of generator to sampling, when \( n \leq 2m \), that is, when resistance to sampling is about \( 2^{2m} \), where \( 2m \) is the total length of irregularly clocked registers. In a modified version of the algorithm, \( T \) random elements of set \( A(Z) \) can be found. But, in this case, the question is how big should be this set to include correct initial state of one of registers. In [14], formulas (2) and (3) determine the probability and the conditional entropy of solutions.

Generally, the reduced complexity attack is to find initial states of Günther generator among a certain set of possible initial states. The most likely solutions are being found using the edit probability calculated for each possible initial state and given segment of output sequence. The time complexity of this attack for a random segment of output sequence with weight of \( w \) and length of \( n \) is \( O(m^22^n) \), where \( m \) is the length of register, which is searched, \( \gamma \) depends on \( m \) and \( \gamma \leq 1 \). For \( \gamma = 1 \) and for \( h(w/n)=2/3 \) the attack is similar to Johansson attack [9] and asymptotic complexities are as follows: time \( O(m^22^{2/3}n) \), memory \( O(2^{2/3}n) \). In comparison with the Johansson attack, the reduced complexity attack is more flexible in terms of useful output sequences, which weights can be freely chosen.

**F. Algebraic attack**

The algebraic attack on stream ciphers with irregularly clocked registers was presented in [13]. The time complexity of the attack on the original ASG generator is \( O(m^21 + m^2)2^n \). In [16,17,18] there were described further similar attacks on modified generators with alternately clocked registers, as MAG or \( 2 \)-MGs. Such attacks have higher time complexity than reduced complexity attacks, but they need shorter, known output sequence. Authors use a linear relationship (XOR) between sequences from registers at the output of the generator and they find sequences of individual clocked irregularly registers by searching among all possible initial states of clock control register.

In the case of the attack on the modified \( k \)-generator of the second type: MAG, for known feedbacks of registers, the attack needs \( k+m_1+m_2 \) bits of output sequence to find the initial states of registers. The time complexity of the attack is then \( O(2^{2l+1}) \). When feedbacks of registers are not known, it must be Berlekamp-Massey algorithm additionally used, hence it is required to know \( k+2m_1+2m_2 \) bits of output sequence to execute the attack and the complexity is \( O((m^2_1+m^2_2)2^{2l+1}) \).

To avoid such attack, the output of the generator should not be defined either by a linear function or by a function that will approximately describe linear relationships between the output of the generator and the outputs of individual registers.

**IV. ALTERNATING STEP GENERATORS WITH SCRAMBLERS**

Alternating step generators such as ASG, MAG, and MASG have linear functions at the output. The analysis, given in section III, suggests a nonlinear transformation in this place. So we proposed a nonlinear multiplicative scrambler [22] as an output function of these generators. As the scrambler, we use the nonlinear feedback shift register [21] with maximal period and linear complexity close to the period. The general scheme of the generator with the scrambler is presented in Fig. 3.

![Figure 3. Alternating step generator with the nonlinear scrambler](image)

The output sequence from the alternating step generator (ASG, MAG or MASG) is applied to the input of the scrambler, where bit after bit is added (mod 2) to its feedback defined by the nonlinear function \( f_n \) in this case.

Attacks on the alternating step generators explore linearity of the transformation at the output and low linear complexity of shift registers. Hence, in the MASG family we proposed some nonlinear functions – ones as nonlinear feedbacks of shift registers, others as filtering or combining functions of linear feedback shift registers.

Known nonlinear feedback shift registers, which give maximal length output sequences, are too short for practical applications. Therefore, MASG and MAG are constructions, which do not ensure sufficient resistance to the attacks. As well as MASG with nonlinear combining function, which gives not random sequence. Hence, the MASG built with linear feedback shift registers and nonlinear filtering functions proves to be the best choice from the MASG family.

**A. MASG \( 1 \) with the nonlinear scrambler**

The scheme of the MASG \( 1 \) with the nonlinear scrambler [22] is presented in Fig. 4. Controlling (LFSR\( 0 \)) and controlled (LFSR\( 1 \) and LFSR\( 2 \)) shift registers have linear feedbacks and are equipped with nonlinear, 8 input bit Boolean filtering functions \( g_{h_0}, g_{h_1} \) and \( g_{h_2} \) [20].

![Figure 4. MASG\( 1 \)](image)
Controlling register (LFSR₀) and nonlinear feedback shift register (NLFSR) are clocked regularly. Controlled registers (LFSR₁ and LFSR₂) are clocked alternately. Lengths of LFSR₀, LFSR₁, LFSR₂ and NLFSR are \( k=127 \), \( m₁=131 \), \( m₂=137 \) and \( n=31 \), respectively.

MASG₁₅ requires 426 bits for initial states of registers. The key for contemporary stream ciphers should be in range 160-256 bits. So we proposed in [22] the initializing method for the state registers of LFSR₀, LFSR₁, LFSR₂ and NLFSR using the key and the generator.

We experimentally checked randomness of keystreams produced by alternating step generators: ASG, MAG and MASG with the nonlinear scrambler: i.e. ASG₂₅, MAG₂₅ and MASG₂₅. We took 10 GB samples of sequences produced by these three generators starting from randomly selected initial states. We tested the randomness using seven basic statistical tests: frequency, serial, two bit, 8-bit poker, 16-bit poker, runs and the autocorrelation test [23]. Tests results were what we expected for random sequences (see [22] for details).

V. CRYPTOANALYSIS OF THE ASG AND MASG₁₅

Recommended in [23] register lengths for the alternating step generator are about 128. Assume that lengths of LFSR₀, LFSR₁ and LFSR₂ registers are \( k=127 \), \( m₁=131 \) and \( m₂=137 \). Let state register of the NLFSR scrambler has length \( n=31 \) and nonlinear filters of LFSRs in MASG₁ have the nonlinear order equal to 7. We consider key lengths between 128 and 256 bits.

Suppose that the adversary knows the plaintext as well as the construction of the generator, in particular feedbacks of registers and member functions. The aim of the adversary is to find initial states of generator registers in order to reconstruct the secret key.

The basic principle in cryptanalysis is that an effective attack on the cipher should have a lower computational complexity (generally time complexity) than brute force attacks. If such attack exists, the cipher is defined as broken.

A. Divide-and-conquer attack

The time complexity of the divide-and-conquer attack on the original ASG (if one knows feedbacks of registers) is \( O(\min(m₁,m₂)2^4) \). For assumed register lengths, the complexity is \( O(\min(131,137)2^{27}) \approx 2^{34} \). Hence, it follows that the LFSR₀ length in the ASG should be close to the length of the key to avoid this attack.

The time complexity of the divide-and-conquer attack on the MASG₁₅ additionally depends on the length of the NLFSR and the nonlinearity of functions used as filters for LFSRs. For assumed parameters the complexity of the attack is \( O(\min(131, 137)2^{27}2^{31}) \approx 2^{93} \). In this formula we replace lengths of the LFSR₁ and the LFSR₂ with linear complexities of these registers with nonlinear filters. Open problem is how to check that found initial states of the LFSR₀ and NLFSR are proper. Apart from that, to achieve resistance to this attack for key longer than 195 bits, registers should be longer or the nonlinearity of the LFSR filters higher.

B. Edit distance correlation attack

The time complexity of the edit distance correlation attack for alternately clocked registers in the ASG is \( O((m₁+m₂)2^{m₁+m₂}) \). The third register can be restored with the complexity \( O(2^{0.271}) \). For assumed register lengths, it is \( O((131+137)2^{131+137}) \approx 2^{284} \) and \( O(2^{0.271/2}) \approx 2^{34} \). From here, we conclude that the ASG and MASG₁ with or without the scrambler and nonlinear filters are resistant to this attack for assumed parameters and key lengths.

C. Edit probability correlation attack

The time of the edit probability correlation attack for alternately clocked registers in the ASG is \( O(\max(m₁,m₂)2^{\max(m₁,m₂)}) \). For assumed register lengths, it is \( O(\max(131,137)2^{\max(131,137)}) \approx 2^{251} \). Hence, these register lengths are enough for key length 128 bits. For longer keys alternately clocked registers should be longer – close to the key length, taking into account the factor \( O(\max(m₁,m₂)) \).

The time complexity of the edit probability correlation attack on the MASG₁₅ additionally depends on the length of the NLFSR and the nonlinearity of functions used as filters for LFSRs. Filtering functions increase linear complexities of sequences produced by LFSR₁ to \( \begin{pmatrix} 131 \\ 7 \end{pmatrix} \) and by LFSR₂ to \( \begin{pmatrix} 137 \\ 7 \end{pmatrix} \). This causes the complexity of the attack is much higher than the complexity of the brute force attack. Hence, we conclude that the MASG₁ with or without the scrambler is resistant to this attack.

D. Reduced complexity correlation attack

The optimal time complexity of the reduced complexity correlation attack on the ASG is \( O(m^22^{2^3m}) \) and requires \( O(2^{2^3m}) \) bits of sequence, where \( m \) is the length of the register, which is searched: \( m₁ \) or \( m₂ \). For LFSR₀, the complexity is \( O(131^22^{2^3×131}) \approx 2^{101} \) and for LFSR₂ \( O(137^22^{2^3×137}) \approx 2^{105} \). We do not need LFSR₀ to carry out the attack, but we need it to reconstruct the whole key. Taking this into accounts, for the key distributed to all registers proportionally in the initialization phase, we may conclude that each register length in the ASG should be close to the length of the key to avoid this attack.

Filtering functions used in MASG₁ increase linear complexities of sequences produced by alternately clocked LFSRs to \( \begin{pmatrix} 131 \\ 7 \end{pmatrix} \) and \( \begin{pmatrix} 137 \\ 7 \end{pmatrix} \). Hence the complexity of the attack is much higher than the complexity of the brute force
attack. NLFSR additionally protects the generator against known plaintext attack.

E. New reduced complexity attack

The analysis for this attack is the same as for the reduced complexity attack carried out in the section D.

F. Algebraic attack

The time complexity of the algebraic attack on the original ASG is \(O((m_1^3 + m_2^3)2^k)\). For assumed register lengths, the complexity is \(O((131^3 + 137^3)2^{127}) \approx 2^{199}\) and depends mainly on the length of the LFSR. To make the ASG resistant to this attack, LFSR\(_0\) may be slightly shorter than the key, taking into account the factor \(O(m_1^3 + m_2^3)\).

To avoid the algebraic attack, we add NLFSR at the output of the MASG, and nonlinear filters to each LFSR. For assumed register lengths, the time complexity of the attack on the MASG\(_{1S}\) is \(O\left(\frac{131^3}{7} + \left(\frac{137^3}{7}\right)2^{127}2^{31}\right) \approx 2^{270}\), which is higher than the complexity of the brute force attack for 256 bit key.

VI. SUMMARY

The first conclusion from the analysis of the attacks on the alternating step generators was to replace linear feedback shift registers with nonlinear ones. But, we are not able generate longer registers than 31 with full period (max 34 we can find in the literature). So we propose to add nonlinear filters to linear registers. Further analysis led us to add a nonlinear scrambler at the output of the generator. In this way we constructed the modified alternating step generator MASG\(_{1S}\).

In this paper we expanded the analysis of known attacks on the original alternating step generator ASG and the modified alternating step generator MASG\(_{1S}\) in order to resist these generators to the attacks with the complexity better than the brute force. General conclusions derived from this analysis are:

1. The length of each LFSR in the original ASG should be close to the length of the key. The key should be distributed and used in stream ciphers to efficiently protect confidentiality of data transmitted in telecommunication channels.

REFERENCES