A Novel Stream Cipher Based on Nondeterministic Finite Automata

Ghassan Khaleel, Sherzod Turaev
Faculty of Information and Communication Technology
International Islamic University Malaysia
53100 Kuala Lumpur, Malaysia
sherzod@iium.edu.my

Tamara Zhukabayeva
Faculty of Information Technology
L.N. Gumilyov Eurasian National University
010008 Astana, Kazakhstan
tamara_kokenovna@mail.ru

***Abstract*- The modified Dömösi's cryptosystem [1] is a stream cipher based on deterministic finite automata without outputs for encoding and decoding. This cryptosystem uses an additional control system to improve the performance to a better linear time without backtracking. This paper, we propose a nondeterministic variant of the cryptosystem, which allows to reduce the dependency of the key automata on a large size and on reversibility of automata.**

Keywords- stream cipher; finite automata; control system; performance analysis; security analysis.

1 INTRODUCTION

Cryptography is a method or mechanism of storing and transmitting information in a particular form through insecure channels. The main objective of the cryptography is satisfying confidentiality, authentication, data integrity and non-repudiation. The confidentiality means that the information we have in database or in the systems should be out of hands of unauthorized users. The basic element of protecting information confidentiality is encryption. Encryption ensures that only the authorized people can read the information. Whereas, the authentication verifies the users identity before revealing sensitive information. The data integrity is one of the most important factor in the cryptography, it means that the received message should be exactly the sent message. A checksum or hash function would be used to detect corruption errors and estimate overall data integrity. Non repudiation is the ability of a system to confirm that a sender cannot refuse having sent message. On the other hand, cryptographic systems can be classified into two types: symmetric-key and asymmetric-key cryptosystems. Symmetric-key cryptosystems use a single key that both sender and recipient have, whereas public-key systems use two keys, a public key known to everyone and a private key that only the recipient of the messages uses. Symmetric-key systems can be also broadly classified into: block ciphers and stream ciphers. The basic idea of a block cipher is to break a plaintext into fixed length blocks, and encrypt each block separately. A stream cipher encrypts a sequence of data, typically, a bit or byte, by using sequence of keys.

2 RELATED WORKS

In 2008, Dömösi [2-4] proposed a new stream cipher based on Rabin-Scott model of automata (i.e., finite automata without outputs), which act as a key for encrypting plaintexts and decrypting ciphertexts. In this way, Dömösi's cryptosystem is similar to Mealy machine: the encoding and decoding are performed using the same key automaton, but it is different from Mealy machine in generating ciphertext: it does not generate the ciphertext by combining the plaintext bit stream a random bit stream using the exclusive OR operator. Dömösi's cryptosystem overcomes many drawbacks of the automata based cryptosystems mentioned above. Firstly, the random number generator is independent from the key. Secondly, the weakly reversibility of automata does not affect the cryptosystem, so this system cannot be attacked with methods used for defeating FAPKC cryptosystems [5-10]. Thirdly, the key automaton is chosen randomly from a large set of automata with more than 256 states and more that 256 input signals, i.e., more than $(256!)^{256}$ possible key automata to be randomly generated. Thus, it gives a lot of options for choosing the key automaton. It is obviously impossible to break the system using brute-force approach. In addition to those advantages, it can implement the cryptosystem in the software and hardware efficiently due to the simplicity of the operations used. Moreover, Dömösi's cryptosystem overcomes some complicated mechanisms in broadcasting/datacasting systems, i.e., this cryptosystem makes frequent key changes

unnecessary and it also makes possible to start decoding at any time during service provision (i.e. not only at the beginning) [4]. However, Dömösi's cryptosystem suffers from the practical difficulties in the encryption algorithm, which affects the entire performance of the cryptosystem. In order to solve these difficulties, Dömösi proposed some modifications in the encryption process with appropriate type of key automata. Comparing with some stream ciphers, the proposed Dömösi cryptosystem is rather slow. In the security level, the resistance against attacks depends on the construction of large minimal and maximal block lengths of ciphertexts, which results in producing much longer ciphertexts than given plaintexts. Hence, this expansion in the ciphertext may affect the performance of encryption and decryption algorithms.

In order to overcome the drawbacks and improve the performance of Dömösi's cryptosystem to a better linear time without backtracking, G. Khaleel at al. [1] proposed an additional control system used together with the Dömösi's encryption algorithm. This control system prevents backtracking in the encryption algorithm by generating two vectors according to the current state, input signals and final states. The control system consists of the initialization stage and the operation stage. In the initialization stage, the control system generates all the control vectors V_1 and V_2, where V_1 consists of all input signals that take the automaton from the current state to any non-final state, whereas V_2 consists of all input signals that take the automaton from any state to one of the target final states. In the operation mode (Figure1). First, the algorithm constructs a prefix of ciphertext of length $t-1$ by randomly selecting signals from vectors V_1, and second, it selects a random signal from V_2 finalizing the construction of ciphertext. Since the modification overcomes the backtracking, the ciphertext is constructed in linear time proportional to the maximum length of the ciphertext blocks.

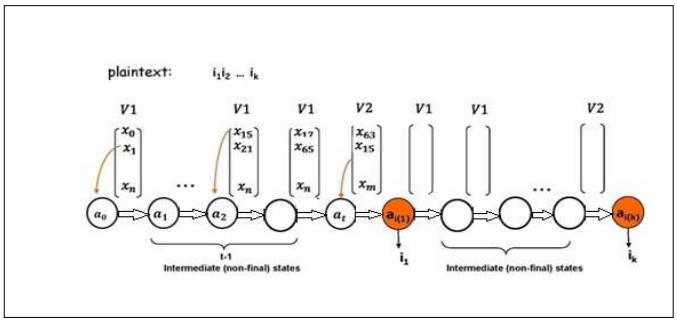

Figure 1. The operation stage

Regarding to the security level of modified Dömösi's cryptosystem, this system has the appropriate avalanche rate and the statistical analysis showed that byte sequence of the encrypted plaintext is random [1].

3 NONDETERMINISTIC AUTOMATA AS A MAIN MODEL

To reduce the dependency of key automaton on the size and reversibility of automata, and enhance the security level, we introduce a new stream cipher based, replacing as deterministic finite automata in the system with their nondeterministic counterparts. In nondeterministic finite automata model (NFA), we can use same ciphertext signals to increase numbers of ciphertext blocks, hence increasing the system immunity against many types of attacks such as brute-force attack. However, we cannot directly use the nondeterministic model as a key for encryption and decryption, due to "nondeterminism". Because, in the decryption process, the key automaton cannot uniquely define the next state to move. As it is illustrated in the state diagram in Figure 2, the signal x_0 can take the automaton to different states.

Formally, an NFA is 5-tuple $M=(A,a_0,\delta,\Sigma,F)$ where A is a finite non-empty set of states, Σ is an alphabet of input signals, $a_0 \in A$ is the initial state, $F \subseteq A$ is a set of final states, and $\delta : A\times\Sigma \to P(A)$ is a transition function with the power set $P(A)$ of A.

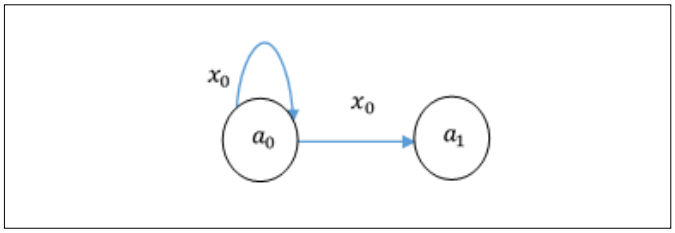

Figure 2. The state diagram of an NFA

To use NFA as a key automaton in the encryption and decryption algorithms, we propose the following modified procedure:

1. Let $M=(A,a_0,\delta,\Sigma,F)$ be a nondeterministic automaton. For each signal x in the ciphertext alphabet Σ corresponding to current state $a \in A$, we assign an extra control information $i_j \in \{0,1\}^*$ as shown in the Figure 3.

2. To generate these information i_j, we introduce the following procedures:

   - Let a function $\mu : A\times\Sigma \to$ takes two parameters: the current state a and signal x as an input, and the output parameter is the number of the next state(s). For example, let $\delta(a,x)=\{a_1,a_2\}$, where $a_1,a_2 \in A$, then $\mu(a,x)=2$.
   - If $\mu(a,x)=1$ then $i_0=0$
   - If $\mu(a,x)=2$ then $i_0=0$, $i_1=1$ etc.

3. We propose a new transition function δ^* takes the key automaton from the current state a to the next state(s) such that $\delta^*(a, x, i_j) \to 2^A$, where the new transition function δ^* takes i_j as an additional argument together with the current state a and ciphertext signal x.

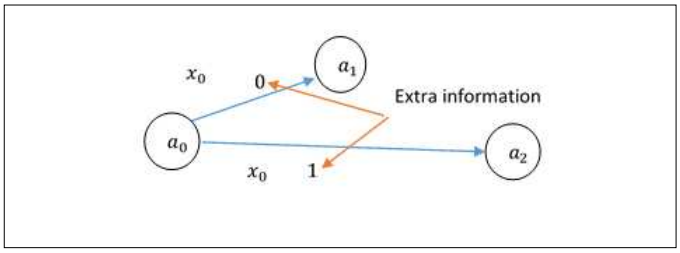

Figure 3. A control information

4. In order to select an extra information i_j in the encryption and decryption algorithms, we use the same pseudo-random seeds during the encryption and decryption processes, otherwise, the decryption does not produce the same plaintext, and set $i_j = PRNG() \bmod \mu(a, x)$, where a and x are current state and input signal respectively.

New encryption and decryption algorithms based on nondeterministic automata and control system approach are illustrated in the Algorithms 1 and 2.

Algorithm 1: Encryption algorithm

Procedure **MODIFIEDENCRYPTION**
Input: $b_1 b_2 \cdots b_n \in \prod^+$
Output: $w_1 w_2 \cdots w_k \in \Sigma^*$
$p \leftarrow a_0, \quad i \leftarrow 1;$
while $i \le k$ **do**
read b_i;
$w_i \leftarrow \lambda;$
select a random t **with** $s_{\min} \le t \le s_{\max}$;
$j \leftarrow 0;$
while $j \le t-1$ **do**
select a random $x \in V_1[p]$
select a random r **with** $\overrightarrow{\delta^\wedge}(p, w_i x, r)$
and $a \notin \phi^{-1}(b_i)$;
$w_i \leftarrow w_i x;$
$p \leftarrow a;$
$j \leftarrow j+1;$
select $x \in V_2[b_i][p];$
select a random r **with** $\overrightarrow{\delta^\wedge}(p, w_i x, r)$
and $a \in \phi^{-1}(b_i)$;
$w_i \leftarrow w_i x;$
$p \leftarrow a;$
$i \leftarrow i+1;$
return $w_1 w_2 \cdots w_k$;

Algorithm 2: Decryption Algorithm

Procedure **MODIFIEDDECRYPTION**
Input: $x_1 x_2 \cdots x_k \in \Sigma^*$
Output: $b_1 b_2 \cdots b_n \in \prod^+$
$p \leftarrow a_0, \quad i \leftarrow 0; \; j \leftarrow 0;$
while $i \le k$ **do**
select a random number r;
$a \leftarrow \delta^*(p, x_i, r);$
$j \leftarrow j+1;$
if $(a \in F \;\&\; j \ge s_{\min})$ **do**
$j \leftarrow 0; \; i \leftarrow i+1;$
$b_i \leftarrow \phi(a);$
return $b_1 b_2 \cdots b_n \in \prod^+;$

4 EXAMPLE

We consider a small key automaton for our proposed cryptosystem. Let $M = (A, a_0, \delta^*, \Sigma, F)$ be an automaton, where $\Sigma = \{x_0, x_1, x_2, x_3, x_4\}$, and $F = \{a_3, a_4\}$ such that $\phi^{-1}(b_1) = a_3$, where $b_1 \in \prod$ is a plaintext character and the transition matrix δ^* with the extra information are defined as shown in the Figure 4, and let the length of the ciphertext block is 4.

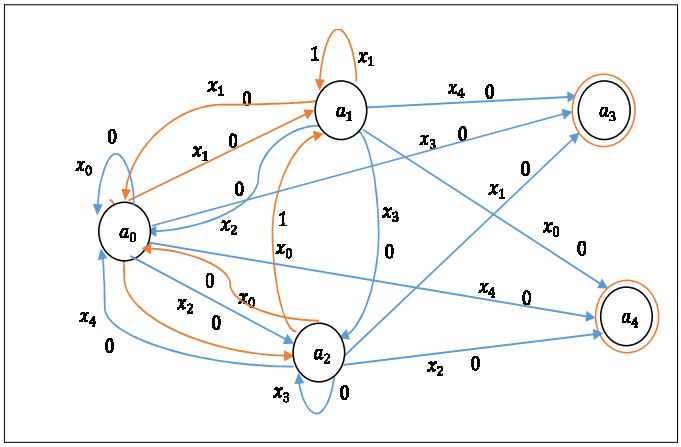

Figure 4. A nondeterministic finite automaton with control information

Consider the generation of the ciphertext w_1 corresponding to the plaintext character b_1. Based on the control system approach, we define two vectors V_1 and V_2 for every state, for instance,

$$V_1[a_0] =< x_0, x_1, x_2 >$$

$$V_1[a_1] =< x_1, x_2, x_3 >$$

$$V_1[a_2] =< x_0, x_3, x_4 >$$

$$\dots$$

$$V_2[a_0, b_1] =< x_3 >$$

$$V_2[a_1, b_1] =< x_4 >$$

$$V_2[a_2, b_1] =< x_1 >$$

$$\dots$$

First, the encryption algorithm constructs a prefix of ciphertext w_1 by randomly selecting signal x_i from vectors V_1, and at the same time it generates an extra information i_j for every signal x_i by using pseudo-random number generator. This process is repeated three times, due to the length of ciphertext block for each plaintext symbol is four. Finally, it selects a signal from V_2 based on the current state and the final state of b_1, finalizing the construction of ciphertext w_1. Let $\{x_2, x_0, x_3, x_1\}$ and $\{0,1,0,0\}$ be the ciphertext characters and extra information, respectively, that randomly selected from V_1, V_2 and pseudo-random number generator. Then, the output ciphertext corresponding to the plaintext b_1 is $w_1 = x_2 x_0 x_3 x_1$. The Figure 5 shows producing the ciphertext based on the control system and the nondeterministic finite automaton.

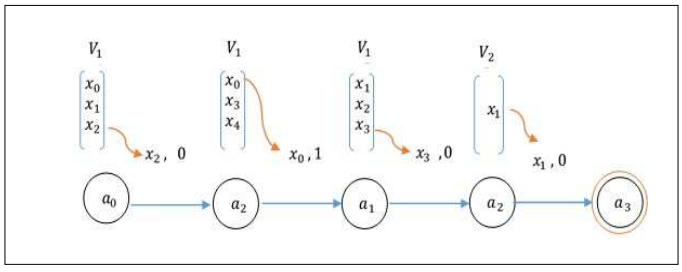

Fig. 5. A ciphertext generation

In the decryption algorithm, the procedure reads the ciphertext characters $\{x_2, x_0, x_3, x_1\}$, at the same time the pseudo-random number generator produces same sequence of the extra control information $\{0,1,0,0\}$ corresponding to the ciphertext characters, then the key automaton goes into states $\{a_0, a_2, a_1, a_2, a_3\}$ under the effect of $\{x_2, x_0, x_3, x_1\}$ and $\{0,1,0,0\}$, hence the state a_3 is the final state, and b_1 is recovered plaintext character, where $\phi(a_3) = b_1$.

5 Advantages of the Proposed Cryptosystem

In the Dömösi's cryptosystem, and also in the modified Dömösi cryptosystem, the only way of resisting against various types of attacks is to apply large automata, and relatively large numbers for minimal and maximal length of ciphertext blocks. Based on Dömösi's principle, for every plaintext character, there is at least m^{k-1} ciphertext blocks, where k is the average length of the ciphertext blocks and m is number of signals that take the key automaton to non-final states from the initial state or a final state. For example, let $m = 128$ and $k = 5$. Then number of ciphertext blocks corresponding to every plaintext character is $128^4 = 2^{28}$. To increase numbers of ciphertext blocks, Dömösi proposed to increase the length of ciphertext block, for instance, if $k = 7$, then, the number of the ciphertext blocks becomes $128^6 = 2^{42}$. While, in our proposed system we can increase the numbers of the ciphertext blocks by increasing numbers of signals, based on the formula: $(n \times m)^{k-1}$, where n is the number of extra signals in the key automaton and $n \geq 2$. Let $n = 8$ and $k = 5$, then the number of the ciphertext blocks becomes 2^{50}, hence it improves the cryptosystem immunity, and at the same time reduce the size of the ciphertext. Moreover, nondeterminism allows to avoid the reversibility of finite automata. Therefore, the cryptosystem based on nondeterministic finite automat cannot be attacked using reversibility.

6 Experimental Tests

To estimate the security and performance of proposed cryptosystem based on nondeterministic finite automata, we consider a large automaton, number of states and ciphertext signals is 256, there are 16 final states, and for each signal take the automaton to non-final state in the key automaton, there are three additional signals $(n = 4)$. The tests have been held in Lenovo Notebook E430 having Intel(R) Core(TM) i5-3230M CPU 2.6 GHz with 4 GB RAM under 64 bit operating system Win10. The simulation programs are compiled using C++.

6.1 Security and performance tests

A strong cipher is capable of resisting against all types of attacks such as statistical, differential, brute-force, known-plaintext, chosen-plaintext and adaptive-chosen plaintext attacks. On the other hand, the computational performance of encryption and decryption schemes is very important. In this

section, we perform a series of tests to estimate the security and performance of the proposed cryptosystem.

6.2 Key Automaton Space Against Cryptanalytic Attacks

Since a key automaton has 256 states and 256 input signals, nondeterminism produces 960 signals $(960 = 4(256-16))$. Therefore, there are more than $(256!)^{960}$ possible key automaton to be randomly generated. It is obviously impossible to break the system using brute-force attack. Moreover, if the average length of the ciphertext blocks is 27, the expected number of encoded messages for each plaintext character is approximately 2^{257}. So, the ciphertext-only attack, or even the known-plaintext attack cannot break the system, since this information cannot be useful to identify another encryption of the same message. Moreover, if the attacker choses or even modifies the plaintext, and gets the corresponding ciphertext (adaptive chosen-plaintext attack), again, this information cannot be used to identify another encryption of the same message.

6.3 Statistical analysis

In order to test the randomness of proposed cryptosystem, we test the sequence of stream bytes of the output ciphertext, by using ENT 2004 program. To perform these tests, we use a random sample of plaintext of size 5 KB and about 90 KB size of ciphertext, let the minimal length of the ciphertext block is 9, maximal length of ciphertext block is 10 and for each signal there are several additional signals. Table 1 shows that the output ciphertext has high entropy. So, the information is essentially random. In addition, the arithmetic mean value of the proposed cryptosystem reaches to 127.5, thus the information is close to random. Chi-square distribution test shows that the byte sequences of the ciphertext are random. Moreover, the serial correlation coefficient is close to the zero, which means that the byte sequence of the ciphertext is uncorrelated.

TABLE 1. The randomness test of the proposed cryptosystem

No. of extra signals	***Entropy***	***Chi square***	***Mean value***	***Monte Carlo***	***Serial correlation***
2	7.997814	21.99%	127.48	3.1362	0.0008
3	7.998164	89.10%	127.54	3.1517	0.0026
4	7.997949	49.28%	127.88	3.1496	0.0087
5	7.998037	67.38%	127.88	3.1453	0.0023

Finally, we can say that sequence of stream byte of the output ciphertext is random. Hence, the proposed cryptosystem is strong against statistical attacks.

6.4 Performance Test

In this test, we use the automaton above. Table 2 show the results of performance tests of the proposed encryption and decryption algorithms. The results of the performance tests show that the encrypting time is more than 184 megabytes per second, whereas decrypting time is about 35 megabytes per second.

7 CONCLUSION AND FUTURE WORKS

This paper proposed a novel stream cipher based on nondeterministic finite automata as keys for encryption and decryption. Simple example showed how to use a nondeterministic automaton as a key automaton. While, security and performance analyses proved that a novel cryptosystem is resistant against many types of attacks and has a high performance.

TABLE 2. The results of the performance tests

Plaintext size(MB)	***Encrypting time(sec.)***	***Decrypting time(sec.)***
1	0.006	0.036
2	0.0109	0.065
3	0.0160	0.096
4	0.0219	0.12
5	0.0270	0.14
6	0.0330	0.158
7	0.0379	0.172
8	0.0439	0.193
9	0.0490	0.22
10	0.0540	0.25

References

1 G. Khaleel, S. Turaev, M.I. Mohd Tamrin and I.F. Al-Shaikhli, “A Performance Improvement of Dömösi’s Cryptosystem”, AIP Conference Proceedings 1705, 020007, 2016.

2 P. Dömösi, “A novel cryptosystem based on finite automata without outputs”, In: M. Ito, Y. Koboyashi, and K. Shoji (eds.), Automata, Formal Languages and Algebraic Systems, World Scientific, p. 23-32, 2008.

3 P. Dömösi, “A novel stream cipher based on finite automata”, In: IntelliSec – The 1st International Workshop on Intelligent Security Systems. Bucharest, Romania (November 11-14, 2009).

4 P. Dömösi, P.: US. Pub. No. US 2009/0092251 A1.

5 R. Tao, S. Chen, “A finite automaton public key cryptosystem and digital signature”, Chinese Journal of Computers 8(6), pp. 401–409, 1985.

6 R. Tao, S. Chen, “Two varieties of finite automaton public-key cryptosystem and digital signatures”, J. of Compt. Sci. and Tech. 1, pp. 9–18, 1986.

7 F. Bao, Y. Igarashi, “Break finite automata public key cryptosystem”, In: International Congress of Mathematicians, pp. 147–158, 1995.

8 R. Tao, S. Chen, “FAPKC3: a new finite automaton public key cryptosystem”, Journal of Computer Science and Technology 12(4), pp. 289–305, 1997.

9 R. Tao, S. Chen, “The generalization of public-key cryptosystem FAPKC4”, Chinese Science Bulletin 44(9), pp. 784–790, 1999.

10 R. Tao, “Finite automata and application to cryptography. Springer-Verlag, Berlin Heidelberg, 2008.