STUDIA UNIV. BABEŞ–BOLYAI, INFORMATICA, Volume **LV**, Number 2, 2010

n-QUASIGROUP CRYPTOGRAPHIC PRIMITIVES: STREAM CIPHERS

ADRIAN PETRESCU

Abstract. In this paper we present two new n-quasigroup stream ciphers based on new n-quasigroup encryption scheme. Also, we present a practical implementation of these ciphers that has very good cryptographic properties. The implementation is based on a design concept of mixing two "incompatible" group operations on the set $\mathbb{Z}_{2^8}$.

1. Introduction

Computationally simple but cryptographically strong cryptographic systems have an important role for efficient digital communication tasks. There is a need for simple cryptographic primitives to implement security in an environment having limited storage and processing power.

Quasigroups based ciphers lead to particular simple yet efficient ciphers.

Almost all results obtained in the application of binary quasigroups in cryptology and coding theory to the end of eighties years of the XX-th century are described in [2] and [3]. A short survey of the known results related to the applications of binary quasigroups for constructing authentication codes, ciphers, and one-way functions is presented in [4].

As far as we know, the only attempts to construct n-quasigroup ciphers are our proposals [8] and [9].

In this paper, we propose two n-quasigroup symmetric-key stream ciphers: a self-synchronized stream cipher and a new type of stream cipher, a totally asynchronous stream cipher.

A totally asynchronous stream cipher is a cipher that cannot recover from an error introduced in the process of communication.

Received by the editors: December 15, 2009.

2010 *Mathematics Subject Classification.* 20N15, 94A60, 68P25.

1998 *CR Categories and Descriptors.* C.2.0 [**Computer System Organization**]: Computer-Communication Networks – *General*; E3 [**Data**]: Data encryption.

Key words and phrases. Cryptography, Security, Stream ciphers.

This paper has been presented at the International Conference Interdisciplinarity in Engineering (INTER-ENG 2009), Târgu-Mureş, Romania, November 12–13, 2009.

Although this property can be seen as a disadvantageous one, there are in fact several useful applications of such ciphers provable secure stream cipher that can guarantee data integrity authentication without using Message Authentication Code or Secure Hash Function.

The implementation of these new ciphers is based on a design concept of mixing two "incompatible" group operation on the same set.

This paper is organized as follows. Section 2 presents a short overview of n-quasigroups. In Section 3 we show the cryptographic properties of n-quasigroup string functions. Section 4 describes a 3-quasigroup encrypting scheme. In Section 5 we present an implementation of a 3-quasigroup self-synchronizing stream cipher and a 3-quasigroup totally asynchronous stream cipher. Conclusions are drawn in Section 6.

2. n-quasigroup definitions

Recall several notions and results which will be used in what follows.

We shall denote the sequence $x_m, x_{m+1}, \ldots, x_n$ by $\{x_i\}_{i=m}^n$ or x_m^n. If $m > n$, then x_m^n will be considered empty.

A non-empty set A together with an n-ary operation $\alpha : A^n \to A$, $n \geq 2$ is called **n-groupoid** and is denoted by (A, α). For $n = 2$ we have a **binary groupoid**.

An n-groupoid (A, α) is called an **n-quasigroup** [1] if the equation

$$\alpha(a_1^{i-1}, x, a_{i+1}^n) = b \tag{1}$$

has a unique solution x for any a_1^n, $b \in A$ and every $i \in \mathbb{N}_n = \{1, \ldots, n\}$.

An equivalent definition, known as *combinatorial definition* is: an n-quasigroup is an n-groupoid such that in the equation

$$\alpha(x_1^n) = x_{n+1} \tag{2}$$

knowledge of any n of the arguments x_1^{n+1} specifies the $(n+1)$-th uniquely.

A **primitive n-quasigroup** [8] is an algebra (A, α, α_1^n), $\alpha, \alpha_i : A^n \to A$, $i \in \mathbb{N}_n$ such that the identities

$$\alpha(x_1^{i-1}, \alpha_i(x_1^n), x_{i+1}^n) = x_i \tag{3}$$

$$\alpha_i(x_1^{i-1}, \alpha(x_1^n), x_{i+1}^n) = x_i \tag{4}$$

$i \in \mathbb{N}_n$, are satisfied.

We note that the operations $\alpha, \alpha_1, \ldots, \alpha_n$ are mutually defined:

$$\alpha(x_1^n) = x_{n+1} \Leftrightarrow \alpha_i(x_1^{i-1}, x_{n+1}, x_{i+1}^n) = x_i, \tag{5}$$

$i \in \mathbb{N}_n$.

An n-quasigroup (A, α) yields a primitive n-quasigroup (A, α, α_1^n) called the **corresponding primitive n-quasigroup**: define $\alpha_i : A^n \to A$, $\alpha_i(a_1^{i-1}, b, a_{i+1}^n) = x$, the unique solution of equation (1).

In turn, a primitive n-quasigroup (A, α, α_1^n) yields n-quasigroups (A, α), (A, α_i), $i \in \mathbb{N}_n$.

Let (A, α) be an n-quasigroup and $[f_1^n; f]$ an ordered system of permutations of the set A. We define a new quasigroup operation β on A as follows:

$$\beta(x_1^n) = f^{-1}(\alpha\{f_i(x_i)\}_{i=1}^n) \tag{6}$$

The n-quasigroups (A, α) and (A, β) are called **isotopic** and $[f_1^n; f]$ an **isotopy of** (A, β) **to** (A, α).

The isotopism of n-quasigroups gives us the power to generate a large number of isotopic n-quasigroups.

3. N-QUASIGROUP STRING FUNCTIONS

In this section we show the cryptographic potentials of n-quasigroup string functions, as a new paradigm in cryptography.

To simplify the notation, we shall consider $n = 3$. The generality of results is not affected.

Let $(A, \alpha, \alpha_1, \alpha_2, \alpha_3)$ be a 3-quasigroup and denote by A^+ the set of all nonempty words formed by the elements of A. For each $a_1a_2a_3a_4 \in A^+$ we define six maps $F_i, G_i : A^+ \to A^+$, $i = 1, 2, 3$, as follows:

$$\begin{aligned} &F_1(x_1 \dots x_n) = y_1 \dots y_n, \\ &y_1 = \alpha(x_1, a_1, a_2), \\ &y_2 = \alpha(x_2, a_3, a_4), \\ &y_j = \alpha(x_j, y_{j-2}, y_{j-1}), \text{ if } j > 2; \end{aligned} \tag{7}$$

$$\begin{aligned} &G_1(x_1 \dots x_n) = y_1 \dots y_n \\ &y_1 = \alpha_1(x_1, a_1, a_2), \\ &y_2 = \alpha_1(x_2, a_3, a_4), \\ &y_j = \alpha_1(x_j, x_{j-2}, x_{j-1}), \text{ if } j > 2; \end{aligned} \tag{8}$$

$$\begin{aligned} &F_2(x_1 \dots x_n) = y_1 \dots y_n \\ &y_1 = \alpha(a_1, x_1, a_2), \\ &y_2 = \alpha(a_3, x_2, a_4), \\ &y_j = \alpha(y_{j-2}, x_j, y_{j-1}), \text{ if } j > 2; \end{aligned} \tag{9}$$

$$\begin{aligned} &G_2(x_1 \dots x_n) = y_1 \dots y_n \\ &y_1 = \alpha_2(a_1, x_1, a_2), \\ &y_2 = \alpha_2(a_3, x_2, a_4), \\ &y_j = \alpha_2(x_{j-2}, x_j, x_{j-1}), \text{ if } j > 2; \end{aligned} \tag{10}$$

$$\begin{aligned} &F_3(x_1 \dots x_n) = y_1 \dots y_n \\ &y_1 = \alpha(a_1, a_2, x_1), \\ &y_2 = \alpha(a_3, a_4, x_2), \\ &y_j = \alpha(y_{j-2}, y_{j-1}, x_j), \text{ if } j > 2; \end{aligned} \tag{11}$$

$$(12)\qquad \begin{array}{l} G_3(x_1 \dots x_n) = y_1 \dots y_n \\ y_1 = \alpha_3(a_1, a_2, x_1), \\ y_2 = \alpha_3(a_3, a_4, x_2), \\ y_j = \alpha_3(x_{j-2}, x_{j-1}, x_j), \text{ if } j > 2. \end{array}$$

We call these maps 3-**quasigroup string functions with initial value** (IV) $a_1a_2a_3a_4$.

The maps F_3 and G_3 are generalizations for n-quasigroups of Markovski's binary quasigroup transformations e and d, respectively [6].

The maps F_i and G_i, $i = 1, 2, 3$ have several useful properties for cryptographical purposes.

1. The maps F_i and G_i are permutations on $A^+ : F_iG_i = G_iF_i = 1_{A^+}$ as a consequence of (3) and (4).

2. Each map F_i can lead to a self-synchronizing stream cipher.

For example, let $m = m_1 \dots m_n \in A^+$ be a plaintext $c = F_3(m) = c_1 \dots c_n$ its ciphertext and $c' = c_1 \dots c_{j-1}c'_jc_{j+1} \dots c_n$, $c'_j \in A$ the received text. Then $G_3(c') = m_1 \dots m_{j-1}m'_jm'_{j+1}m'_{j+2}m_{j+3} \dots m_n$ for some $m'_j, m'_{j+1}, m'_{j+2} \in A$. This result follows directly from the definition of G_3.

3. Each map G_i can leads to a totally asynchronous stream cipher.

For example, if we use G_3 as encrypting function and F_3 as decrypting function, then the rest of message after a ciphertext value error is garbled:

$$\begin{array}{l} m'_j = \alpha(m_{j-2}, m_{j-1}, c'_j), \\ m'_{j+1} = \alpha(m_{j-1}, m'_j, c_{j+1}), \\ m'_{j+2} = \alpha(m'_j, m'_{j+1}, c_{j+2}), \\ m'_{j+3} = \alpha(m'_{j+1}, m'_{j+2}, c_{j+3}), \dots \end{array}$$

4. Each map $F_i(G_i)$ can lead to a stream cipher resistive on the brute force attack.

For example, suppose that an intruder knows a cipher text $c = c_1, \dots, c_n = F_1(x_1 \dots x_n)$, where $x_1 \dots x_n$ represents the unknown plaintext. Then, for recovering the quasigroup operation α which is the key of the encrypting method, it should solve a system of equations of the form (7). Taking into account (5), the following statement is true.

Let $c_1 \dots c_n \in A^+$ be a given string. For any 3-quasigroup operation β on A and any elements $a_1, a_2, a_3, a_4 \in A$, there are uniquely determined elements $x_1, \dots, x_n \in A$ such that the equality $F_i(x_1 \dots x_n) = c_1 \dots c_n$ $(G_i(x_1 \dots x_n) = c_1 \dots c_n)$ holds.

Indeed, for example, if $i = 1$ we have

$$c_1 = \beta(x_1, a_1, a_2) \Leftrightarrow x_1 = \beta_1(c_1, a_1, a_2)$$
$$c_2 = \beta(x_2, a_3, a_4) \Leftrightarrow x_2 = \beta_1(c_2, a_3, a_4)$$
$$c_j = \beta(x_j, c_{j-2}, c_{j-1}) \Leftrightarrow x_j = \beta_1(c_j, c_{j-2}, c_{j-1})$$

if $j > 2$.

So, the system $F_i(x_1 \dots x_n) = c_1 \dots c_n$ has as many solutions as there are 3-quasigroup operations on the set A.

If $|A| = m$ (cardinality of A), then there are at least $m!(m-1)! \dots 2!1!$ binary quasigroup operations on A. From each binary quasigroup $(A, \cdot)$ we can derive two 3-quasigroups, $\alpha(x_1, x_2, x_3) = (x_1 \cdot x_2) \cdot x_3$ and $\beta(x_1, x_2, x_3) = x_1 \cdot (x_2 \cdot x_3)$.

Such 3-quasigroups are called **reducible**. But there exist irreducible 3-quasigroups with carrier A. Hence the number of 3-quasigroups (A, α) is very large.

5. If an intruder knows both the plaintext and the corresponding ciphertext, in some cases it can't recover quasigroup operation α (see Section 5).

6. Each map F_i has a nice scrambling property. The following is true.

Let $m = m_1 \dots m_n \in A^+$ be an arbitrary string and let $c = c_1 \dots c_n = F_i(m)$. If n is large enough, then the distribution of elements $c_j, j \in \mathbb{N}_n$ is uniform.

4. A 3-quasigroup encryption scheme

Let $(A, \alpha, \alpha_1, \alpha_2, \alpha_3)$ be a 3-quasigroup called the **seed quasigroup**. Denote by $\mathcal{M}$ the **message space** and $\mathcal{C}$ denotes the **ciphertext space**. We put $\mathcal{M} = \mathcal{C} = A^+$. For each element $a \in A$, let f_a be a permutation of A. $K = A^8 \times \{1, 2, 3\}$ is called the **key space**. An element $k = a_1 a_2 \dots a_8 i$ is called a **key**.

From section 3, it follows that the quasigroup operation α must be kept secret. But is not a good idea to use all the time the same quasigroup. The isotopism of quasigroups gives us the power to use a large number of isotopic quasigroups to seed quasigroup.

To simplify the notation we put $f_j = f_{a_j}$. Using the subkey $a_1 a_2 a_3 a_4$, we define a new quasigroup operation β on A as follows:

$$\beta(x_1, x_2, x_3) = f_4^{-1}(\alpha(f_1(x_1), f_2(x_2), f_3(x_3))).$$

For the 3-quasigroup $(A, \beta, \beta_1, \beta_2, \beta_3)$, consider the quasigroup string function F_i and G_i, $i = 1, 2, 3$, with initial value $a_5 a_6 a_7 a_8$.

Finally, we get two stream ciphers:

- a self-synchronizing stream cipher if for each $i = 1, 2, 3$, F_i is the encryption function and G_i the decryption function.

- a totally asynchronous stream cipher if for each $i = 1, 2, 3$, G_i is the encryption function and F_i the decryption function.

The seed quasigroup $(A, \alpha, \alpha_1, \alpha_2, \alpha_3)$, the key space, the set $\{f_a \mid a \in A\}$ and the definitions of string functions F_i and G_i are public knowledge.

The security of our ciphers lies solely on the key, not on the encryption algorithm. Perfect secrecy in the sense of Shanon is obtained if a "one-time" key is used.

For other arity values of quasigroup operations, the encryption scheme is similar.

5. A practical implementation

This section describes a very fast, strong and small 3-quasigroup self-synchronizing stream cipher and a 3-quasigroup totally asynchronous stream cipher.

From the practical viewpoint, the most important quasigroups are of order 2^8-byte encoding and 2^{16}-word encoding. The usage of a general 3-quasigroup in computation requires to store its Cayley table. For a quasigroup of order n, this table has n^3 elements. In particular $(2^8)^3 = 16$MB. In order to overcome the storage requirements for the Cayley table we consider as seed quasigroup $(\mathbb{Z}_{256}, \alpha, \alpha_1, \alpha_2, \alpha_3)$, $\alpha(x, y, z) = x - y - z \pmod{256}$.

To define permutations f_a, we consider a new group operation $\circ$ on $\mathbb{Z}_{256}$ - multiplication modulo 257. This kind of multiplication was first used in IDEA cipher [5].

To generalize the discussion beyond the case of byte encoding [5], let n be one of the integers 1, 2, 4, 8, 16. As of April 2009 the only know Fermat primes are $2^n + 1$.

Let $(\mathbb{Z}^*_{2^n+1}, \cdot)$ denote the multiplicative group of the field $\mathbb{Z}_{2^n+1}$ and let $(\mathbb{Z}_{2^n}, +)$ denote the additive group of the ring $\mathbb{Z}_{2^n}$. Define the direct map

$$d : \mathbb{Z}_{2^n} \to \mathbb{Z}^*_{2^n+1},\ d(x) = \begin{cases} x, \text{if } x \neq 0 \\ 2^n, \text{if } x = 0 \end{cases},$$

and via d and its inverse d^{-1} define a new binary operation on $\mathbb{Z}_{2^n}$,

$$x \circ y = d^{-1}(d(x) \cdot d(y)).$$

Then $(\mathbb{Z}_{2^n}, \circ)$ is a cyclic group isomorphic to $(\mathbb{Z}^*_{2^n+1}, \cdot)$.

On the set $\mathbb{Z}_{2^n}$ we have two group operations $(\mathbb{Z}_{2^n}, +, \circ)$. These operations are "incompatible" in the sense that:

- no distributive law is satisfied:

$$x \circ (y + z) \neq (x \circ y) + (x \circ z),$$
$$x + (y \circ z) \neq (x + y) \circ (x + z);$$

- no generalized associative law is satisfied

$$x \circ (y + z) \neq (x \circ y) + z,$$
$$x + (y \circ z) \neq (x + y) \circ z.$$

Meier and Zimmerman [7] proposed a good performance algorithm and implementation for multiplication modulo 2^n+1. This algorithm requires a total of six addition and subtractions, one 8 (16) bit multiplication and one comparison and is based on the following result [5]. Let a,b be two n bit non-zero integers in Z_{2^n+1}. Then

$$ab(\mathrm{mod}(2^n+1)) = \begin{cases} ab(\mathrm{mod}2^n) - ab\,\mathrm{div}2^n, & \text{if } ab(\mathrm{mod}2^n) \geq ab\,\mathrm{div}2^n \\ ab(\mathrm{mod}2^n) - ab\,\mathrm{div}2^n + 2^n + 1, & \text{otherwise} \end{cases}$$

where $ab\,\mathrm{div}2^n$ denotes the quotient when ab is divided by 2^n.

Now, for each $a \in \mathbb{Z}_{256}$, define the permutation f_a to be $f_a(x) = x \circ a$.

We define the key space $K = \mathbb{Z}_{256}^9$. For each key $k = a_1 \ldots a_8 a_9$, we have

$$\begin{aligned}
\beta(x_1,x_2,x_3) &= (x_1 \circ a_1 - x_2 \circ a_2 - x_3 \circ a_3) \circ a_4^{-1} \\
\beta_1(x_1,x_2,x_3) &= (x_1 \circ a_4 + x_2 \circ a_2 + x_3 \circ a_3) \circ a_1^{-1} \\
\beta_2(x_1,x_2,x_3) &= (x_1 \circ a_1 - x_2 \circ a_4 - x_3 \circ a_3) \circ a_2^{-1} \\
\beta_3(x_1,x_2,x_3) &= (x_1 \circ a_1 - x_2 \circ a_2 - x_3 \circ a_4) \circ a_3^{-1}
\end{aligned}$$

Hence the decryption is essentially the same process as encryption.

We set $i = 3$ if $a_9 \equiv 0 \pmod 3$ and $i = a_9 \pmod 3$ otherwise.

Therefore, we have 2^{32} 3-quasigroups on $\mathbb{Z}_{2^8}$ and $3 \cdot 2^{32}$ pairs of encryption and decryption functions.

The "incompatibility" of the operations $+$ and $\circ$ implies a strong resistance on known plaintext attack. If an intruder already knows both the plaintext $m = m_1 \ldots m_n$ and the associated ciphertext $c = c_1 \ldots c_n$, as far as we know, brute force is the only method to recover the key from equations of the from

$$c_j = (a_1 \circ c_{j-2} - a_2 \circ c_{j-1} - a_3 \circ m_j) \circ a_4^{-1}$$

for encryption function F_3, for example.

The security of the proposed cipher needs further investigations. The author hereby invite interested parties to attack this proposed cipher and will be grateful to receive the results of any such attacks.

We note that an uniform distribution of the characters of the ciphertext occurred in every of more than 50 experiments, even for short plaintexts.

The cipher was implemented in programming languages $C++$ and Java. In assembly language the obtained code is tiny.

Finally, we present a simple speed test for a $C++$ implementation of this cipher. We compared the average elapsed times in seconds to encrypt and decrypt a file with that to copy the same file one byte at a time.

In a similar way we get a 3-quasigroup totally asynchronous stream cipher. We interchange the maps F_i and G_i.

Table 1. Speed test

File size	File copy	Encrypt	Decrypt
489 KB	0.062	0.094	0.094
1.22 MB	0.170	0.219	0.219
2.11 MB	0.234	0.391	0.391
6.01 MB	0.672	1.141	1.141

6. Conclusions

These ciphers are appropriate for a fast online digital communication.

The ciphers structure facilitate a hardware implementation. The similarity of encryption and decryption makes it possible to use the same device in both encryption and decryption.

An extension to n-quasigroups of the encryption scheme (Section 4) is obvious.

In order to improve the security of the proposed cipher, 3-quasigroups can be replaced by n-quasigroups ($n = 4, 5, \dots$) and/or $\mathbb{Z}_{2^8}$ can be replaced by $\mathbb{Z}_{2^{16}}$.

References

[1] V.D. Belousov, *n-ary quasigroups*, Stiintca, Kishinev, 1972 (in Russian).

[2] J. Dénes, A.D. Keedwell, *Latin Squares. New Developments in the Theory and Applications*, North-Holland Publ. Co., Amsterdam, 1991.

[3] J. Déned, A.D. Keedwell, *Some applications of non-associative algebraic systems in cryptology*, PU.M.A., 12, No.2, 2002, 147-195.

[4] M.M. Glukhov, *Some applications of quasigroups in cryptography*, *Prikl. Diskr. Math.*, No. 2 (2), 2008, pp. 28-32 (in Russian).

[5] X. Lai and J.L. Massey, *A proposal for a new block encryption standard*, *Proc. of EUROCRYPT'90*, 1990, pp. 389-404.

[6] S. Markovski, *Quasigroup string processing and application in cryptography*, Invited talk, Proc 1st International Conference on Mathematics and Informatics for Industry, Thessaloniki, Greece, 2003.

[7] C. Meier and R. Zimmerman, *A multiplier modulo* $(2^n + 1)$, Diploma Thesis, Institut für Integrierte Systems, ETH Zürich, Switzerland, February 1991.

[8] A. Petrescu, *Applications of quasigroups in cryptography*, *Proc. Inter-Eng 2007*, Univ. "Petru Maior" of Tg. Mures, Romania, 2007.

[9] A. Petrescu, *A 3-quasigroup stream cipher*, Proc. Inter-Eng. 2009, Univ. "Petru Maior" of Tg. Mures, Romania, 2009, 264-267.

Department of Mathematics and Informatics, Faculty of Sciences and Letters, "Petru Maior" University of Tg. Mureş, 1 Nicolae Iorga Street, 540088 Târgu-Mureş, Romania

E-mail address: apetrescu@upm.ro