Sequence Randomization Using Quasigroups and Number Theoretic Transforms

Vaignana Spoorthy Ella
Department of Computer Science
Oklahoma State University
Stillwater, Oklahoma, USA.
spoorthy.ella@okstate.edu

Abstract — **This paper investigates the use of quasigroups, Hadamard transforms and Number Theoretic Transforms for use in sequence randomization. This can also be used to generate hash functions. We use autocorrelation and Chi squared tests to test the randomness of the generated sequence.**

Keywords- Encryption, Quasigroup, Hadamard matrix, Number theoretic transforms, Pseudo-random sequences

I. INTRODUCTION

Good symmetric encryption schemes as well as randomization and hashing techniques are based on effective techniques of confusion and diffusion [1]. Quasigroups provide an excellent way to generate an astronomical number of keys and therefore they are excellent at confusion [2] but they are not equally good at diffusing the statistics of the plaintext. Specifically, the quasigroup transformation can be easily discovered by the known plaintext attack. For quasigroup mappings in encryption, it is necessary, therefore, to use this mapping together with other statistics-diffusing mappings.

A quasigroup $(Q, *)$ is a set of numbers with a binary operation * such that for each *a and b* in Q, there exist unique elements x and y in Q such that:

$$a*x = b \text{ and } y*a = b$$

In other words, for two elements a and b, b can be found in row a and in a column b of the quasigroup table of element operations. The group operations thus are equivalent to a table of permutations.

A. Example

A quasigroup of order 7 consisting of 0, 1, 2, 3, 4, 5, 6
Index

*	0	1	2	3	4	5	6
0	2	1	0	5	4	6	3
1	1	4	3	2	0	6	5
2	0	5	1	6	3	4	2
3	4	3	6	1	2	5	0
4	6	2	5	0	1	3	4
5	3	0	2	4	5	1	6
6	5	6	4	3	0	2	1

Suppose a=6 and x=3 and, then in $a*x = b$, b=3.

B. Encryption using quasigroup

Consider $(a_1, a_2, a_3...a_n)$ are input elements; a is the seed element, and $(e_1, e_2, e_3...e_n)$ are output elements after the quasigroup transformation.

$$e_1=a*a_1$$
$$e_2=e_1*a_2$$
$$e_3=e_2*a_3$$
$$.$$
$$.$$
$$.$$
$$e_n=e_{n-1}*a_n$$

C. Example

Seed element a=3

a_1	a_2	a_3	a_4	a_5	a_6
4	5	6	0	1	3

e_1	e_2	e_3	e_4	e_5	e_6
2	4	4	6	6	3

$$e_1=a*a_1=3*4=2$$
$$e_2=e_1*a_2=2*5=4$$
$$e_3=e_2*a_3=4*6=4$$
$$e_4=e_3*a_4=4*0=6$$
$$e_5=e_4*a_5=6*1=6$$
$$e_6=e_5*a_6=6*3=3$$

One way to diffuse statistics effectively is the use of transforms [3] where the security is enhanced by using a variety of them and by chaining them [4]. Here we investigate the use of chained Hadamard transforms and NTTs (number theoretic transforms) to introduce diffusion together with the quasigroup transformation.

Number Theoretic Transforms are also a type of discrete Fourier transforms. The N*umber theoretic transform* is based on generalizing the nth primitive root of unity to a quotient ring instead of using complex numbers.

In this paper, the input sequence will undergo different transformations sequentially like quasigroup transformation, Hadamard transformation and Number theoretic transformation. For Hadamard and Number theoretic transforms, the input is split into certain group of bits such that each group bit count is the order of the corresponding matrix.
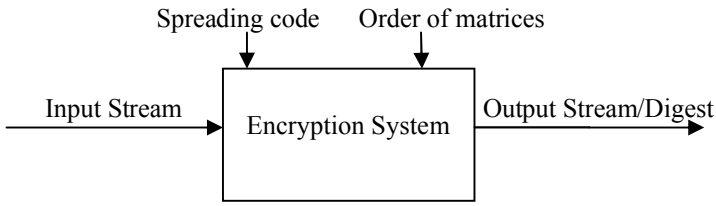
Figure 1 The General architecture of the proposed encryption and hashing system

II. HADAMARD TRANSFORMS

The Hadamard transform is a generalized class of discrete Fourier transforms [4], [18], [19].It is generated either recursively, or by using binary representation. All the values in the matrix are non-negative. Each negative number is replaced with corresponding modulo number. For example in modulo 7 Hadamard matrixes -1 is replaced with 6 to make the matrix non-binary. Due to its symmetric form it can be used in applications such as data encryption and randomness measures [5]-[8]. Only prime modulo operations are performed because non-prime numbers can be divisible with numbers other than 1 and itself [2].Recursively, we define the 1×1 Hadamard transform H_0 by the identity $H_0 = 1$, and then define H_m for $m > 0$ by,

$$H_m = \frac{1}{\sqrt{2}} \begin{pmatrix} H_{m-1} & H_{m-1} \\ H_{m-1} & -H_{m-1} \end{pmatrix}$$

A Hadamard matrix, $\mathbf{H}_n$, is a square matrix of order n = 1, 2 or 4kwhere k is a positive integer. The elements of $\mathbf{H}$ are either +1 or −1 and $\mathbf{H}_n.\mathbf{H}_n^T = n\mathbf{I}_n$, where $\mathbf{H}_n^T$ is the transpose of $\mathbf{H}_n$, and $\mathbf{I}_n$ is the identity matrix of order n. A Hadamard matrix is said to be normalized if all of the elements of the first row and first column are +1. Some examples of the Hadamard matrices are given below.

$$\mathbf{H}_{0} = +1$$

$$H_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

Hadamard matrix of modulo 31 of size 8*8

1	1	1	1	1	1	1	1
1	30	1	30	1	30	1	30
1	1	30	30	1	1	30	30
1	30	30	1	1	30	30	1
1	1	1	1	30	30	30	30
1	30	1	30	30	1	30	1
1	1	30	30	30	30	1	1
1	30	30	1	30	1	1	30

Hadamard matrix of modulo 7 of size 4*4

1	1	1	1
1	6	1	6
1	1	6	6
1	6	6	1

The idea of encryption is to multiply the decimated input sequence with the non-binary Hadamard matrix in a chained manner block by block. The block size depends upon the size of the Hadamard matrix you have chosen. Input sequence is taken in the form of column matrix.
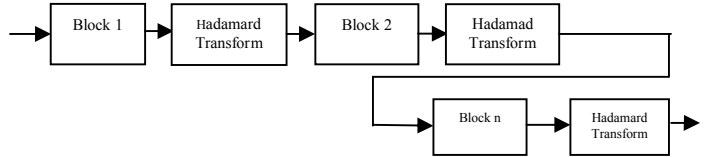

Figure 2 Hadamard encryption

III. NUMBER THEORETIC TRANSFORMS

The *Number theoretic transform* is based on generalizing the nth primitive root of unity to a quotient ring instead of using complex numbers [9].

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & w & w^2 & w^3 \\ 1 & w^2 & w^4 & w^6 \\ 1 & w^3 & w^6 & w^9 \end{bmatrix}$$

The unit w is exp $(2\pi/ n)$. Now everything a number theoretic transform is all about is that $w^n=1$.

NTT matrix of order 6*6

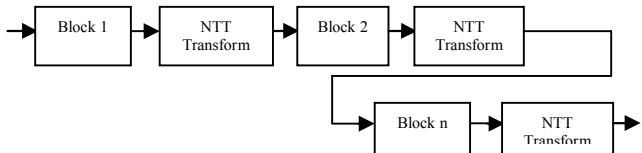| 1	1	1	1	1	1
1	3	2	6	4	5
1	2	4	1	2	4
1	6	1	6	1	6
1	4	2	1	4	2
1	5	4	6	2	3

A. NTT Encryption

Figure 3 Number Theoretic Encryption

The idea of encryption is to multiply the decimated input sequence (which is the output we have obtained after encryption using Hadamard transform with the non-binary Number theoretical matrix in a chained manner block by block. The block size depends upon the size of the Number theoretical matrix you have chosen. The Input sequence is taken in the form of column matrix.

IV. THE PROPOSED ENCRYPTION SYSTEM

A Encryption

- Phase1: Encryption of input data using quasigroup based encryption system.
- Phase2: Output of Phase1 is given as input to the Phase2. In phase2 we perform Hadamard transformation of data.
- Phase3: Output of Phase2 is given as input to the Phase 3. In phase 3 we perform Number Theoretic transform.
- Phase4: Phase2 is repeated with a different order of Hadamard matrix.

B. Decryption

As the Hadamard matrix operations are invertible, we can perform decryption of the data by generating inverse Hadamard matrix.

I/p Integer Stream   **Encrypted o/p**

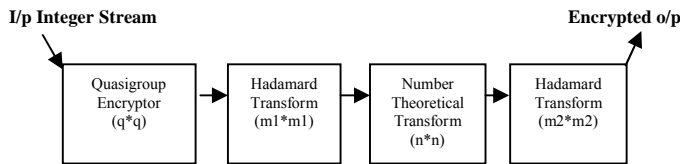| Quasigroup Encryptor (q*q) | → | Hadamard Transform (m1*m1) | → | Number Theoretical Transform (n*n) | → | Hadamard Transform (m2*m2) |

Figure 4   The Proposed Encryption system

All the matrices such as the quasigroup, Hadamard Matrix and Number Theoretic transform matrix should have different orders so that at each encryption level, the size of the input block size differs which eventually increases the randomness of the input sequence. Input size should be multiple of orders of all three matrices i.e. quasigroup, Hadamard and Number theoretic matrix in order to get the suitable block size.
Hadamard transforms and Number Theoretic transforms work as hash functions which generate different hash values for different input values [2].If one bit changes in the input sequence, there is a large difference in the generated random sequence.

V. PERFORMANCE MEASURE OF THE SYSTEM

A. Using Autocorrelation Function.

1. Tests of Randomness
We use autocorrelation functions to test the randomness of input and output data. If the given sequence is random, autocorrelation values should be nearer to zero.

2. Function of Autocorrelation

$$C(k) = \frac{1}{n}\sum_{j=1}^{n}\left(a_j\, a_{j+k}\right)$$

Where n is period and k=0 to n-1.

3. Randomness Measure

Randomness R of a sequence of period n is measured by the following formula. If the given sequence is random, R value should be nearer to 1.

$$R(Sequence) = 1 - \frac{1}{n-1}\sum_{k=1}^{n-1}\left(|C(k)|\right)$$

While calculating autocorrelation of our encrypted system we replace all integers with their corresponding binary values 0s and 1s. And 0s are replaced with -1.

4. Example 1

Here Input size=684, Period=3*684=2052. (Binary form)
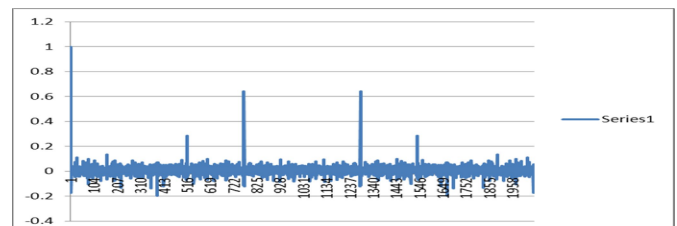Characteristics of input data: Java Pseudo random sequence

Figure 5 Autocorrelation graph for input Pseudo random sequence and R=0.9428.
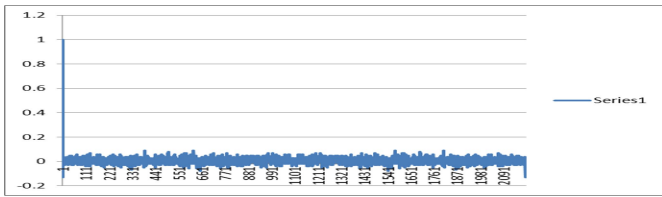
Figure 6 Autocorrelation graph of the output sequence generated by the encryption system R=0.9631

6. Example 2

Input type: d-sequences [11]-[16].

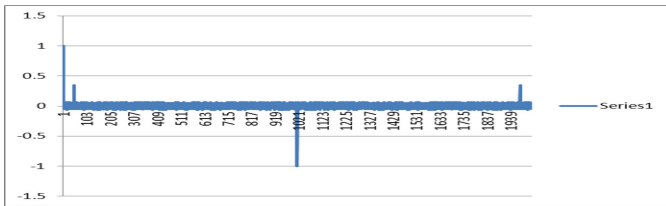Characteristics of d-sequence: prime number used 2029, period: 2028

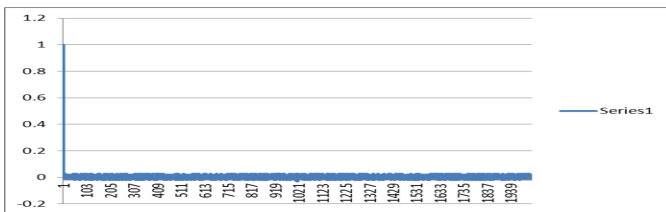Figure 7 Autocorrelation graph for input d-sequence and R=0.9238

Figure 8 Autocorrelation graph for output d-sequence and R=0.9798.

7. Example 3

Input size=684, period=3*684=2052

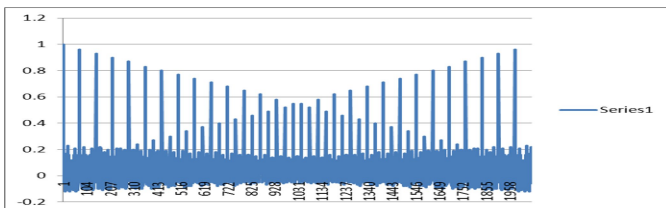Characteristics of input data: All zeros and last bit as '1'

Figure 9 Autocorrelation graph of input sequence with all zeros and last bit as 1 and R=0.9072.
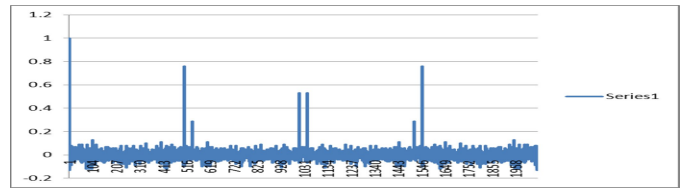

Figure 10 Autocorrelation graph of output sequence. And R=0.9515.

In this case we get output as Pseudo random sequence because we have taken the input with all zeros and the last bit as 1.As a result, quasigroup elements are repeated. Moreover, Hadamard and the Number Theoretic Transforms have not shown any change in the input sequence. We can get better results if we consider quasigroup, Hadamard and Number theoretic matrix having different orders as they are same as block sizes of the transformations.

B. Using Chi Square Statistics

1. Frequency test within a block:

The purpose of this test is find proportion of ones within M-bit blocks. According to this statistics the frequency of ones in an m-bit block is approximately.
M/2 would be expected under an assumption of randomness.

2. Parameters

M= the length of each block.

n= the length of the bit string.

$\chi^2_{(obs)=}$ A measure of how well the observed proportion of ones within a given M-bit block match the expected proportion(1/2).

The reference distribution for the test statistic is a χ^2 distribution.

$\varepsilon=$ the sequence of bits as generated by the random or Pseudo random being tested.

3. Test procedure

Step 1

Partition the input sequence into N= $\lfloor n/M \rfloor$ non-overlapping blocks. Discard any unused bits.

Step 2

 Determine the proportion π_i of ones in each M-bit block using the equation $\pi_i = \dfrac{\sum_{j=1}^{M} \varepsilon_{(i-1)M+j}}{M}$ for $1 \le i \le N$.

Step 3

Compute the χ^2 statistic:

$$\chi^2 \text{(obs)} = 4M \sum_{i=1}^{N} \left(\pi_i - \tfrac{1}{2}\right)^2.$$

Step 4

Compute P-value=igamc(N/2,χ^2(obs)/2),where igamc is the incomplete gamma function.

4. Decision Rule (at the 1% Level)

If the computed P-value is <0.01, then conclude that the sequence is non-random. Otherwise, conclude that the sequence is random.

5. Example 6

Sequence:Output generated using proposed encryption system where input is a pseudo random sequence generated by the Java random function.

Non- binary sequence size: 684
Binary sequence size=684*3=2052

Block Size (M) =18, hence Number of blocks (N) =2052/18=114.

We get χ^2 value as 58
.
P-value= igamc(114/2,58)=0.56977>0.01, hence the generated sequence is random.

6. Example 7

Sequence: Output generated using proposed encryption system where input is a pseudo random sequence generated by the Java random function.
.
Non -binary sequence size: 636
Binary sequence size=636*3=1908

Block Size (M) =18, hence Number of blocks (N) =1908/18=106.

We get χ^2 value as 52.

P-value= igamc(106/2,52)=0.46320>0.01, hence the generated sequence is random.

The above results show that the generated output sequence passes the Chi Square tests of randomness.

VI. CONCLUSION

We have shown that the encryption system which consists of sequential transforms like quasigroup scrambling, Hadamard transform and Number Theoretic transform provides a method of generating random output sequence and also can act as hash function generator. The randomization obtained is very good, and, therefore, one can foresee practical applications for this method.

REFERENCES

[1] C.E. Shannon, Communication theory of secrecy systems. Bell System Technical Journal, 28:656-715, 1949.
[2] M.Satti and S.Kak, Multilevel indexed quasigroup encryption for data and speech.IEEE Trans on Broadcasting 55: 270-281, 2009.
[3] J. Hoffstein, J. Pipher, J.H. Silverman, An Introduction to Mathematical Cryptography. Springer,2010.
[4] R.S.Reddy,Encryption of binary and non-binary data using chained Hadamard transforms.arXiv:1012.4452
[5]V.Godavarty,Using Quasigroups for Generating Pseudorandom Numbers.arXiv:1112.1048
[6]B.Goldburg, S.Sridharan, E.Dawson, Design and cryptanalysis of transform-based analog speech scramblers.Journal of Selected Areas in Communications 11:735-744, 1993.
[7]V.Milosevic, V.Delic, V.Senk,Hadamard transform application in speech scrambling. Proceedings 13th Intl. Conference on Digital Signal Processing, 361-364, 1997.
[8]C. C. Gumas, A century old fast Hadamard transform proves useful in digital communications, Chip Centerquestlink, 2006.
[9]S. Kak, Classification of random binary sequences using Walsh-Fourier analysis.IEEE Trans on Electromagnetic Compatibility EMC-13, pp. 74-77, 1971.
[10]S. Kak, Joint encryption and error-correction coding, Proceedings of the 1983 IEEE Symposium on Security and Privacy, pp. 55-60 1983.
[11]T. Koshy, Elementary Number Theory with Applications, 2nd Ed, Elsevier Inc., Academic Press Publications, Burlington, MA, 2007, pp. 346.
[12]A.Rukhin et al, A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications.National Institute of Standards and Technology Gaithersburg, April 2010.
[13]S. Kak and A. Chatterjee, On decimal sequences. IEEE Transactions on Information Theory, IT-27: 647 – 652, 1981.
[14]S. Kak, Encryption and error-correction coding using D sequences. IEEE Transactions on Computers C-34: 803-809, 1985.
[15]S. Kak, New results on d-sequences. Electronics Letters 23: 617, 1987.
[16]D. Mandelbaum,on subsequences of arithmetic sequences. IEEE Trans on Computers 37: 1314-1315, 1988.
C-34: 803-809, 1985.
[17]S Kak, Prime reciprocal digit frequencies and the Euler zeta function. arXiv:0903.3904