Block Cipher Based on Randomly Generated Quasigroups

Deepthi Haridas, S Venkataraman, Geeta Varadan

Abstract**—**Quasigroups are algebraic structures closely related to Latin squares which have many different applications. The construction of block cipher is based on quasigroup string transformation. This article describes a block cipher based Quasigroup of order 256, suitable for fast software encryption of messages written down in universal ASCII code. The novelty of this cipher lies on the fact that every time the cipher is invoked a new set of two randomly generated quasigroups are used which in turn is used to create a pair of quasigroup of dual operations. The cryptographic strength of the block cipher is examined by calculation of the xor-distribution tables. In this approach some algebraic operations allows quasigroups of huge order to be used without any requisite to be stored.

Keywords**—**quasigroups, latin squares, block cipher and quasigroup string transformations.

I. Introduction

EVEN THOUGH quasigroups(or Latin squares) are used in design of many modern symmetric cyprotgraphic algorithms, the theory and practice of cryptographic quasigroup transformation is still in its early stage. An excellent introductory material about theory of quasigroups can be found in [1]-[4] and there are some applications of quasigroups and Latin squares in [5]-[6].

In the course of the last two decades several cryptographic algorithms have been developed based on quasigroups [5], [7]-[10]. For every pair of quasigroups the attack with differential cryptanalysis on quasgroup block cipher is inefficient [11], where analysis is done by calculation of xor-distribution table. It was shown that as the number of rounds increases the xor-distribution table converges to uniform distribution thereby making the attack with differential cryptanalysis inefficient. Quasigroups are used to define so called "quasigroup string transformation" as reported by many in literature [6]-[7], [11]-[13].

Gligoroski and Markoviski [14] have successfully implemented a block cipher based on quasigroup where the key for the cipher is the triplet 2 quasigroups and the password. There they have fixed the two quasigroup i.e. the quasigroups are public, only password is secret. As an extension to this idea, in the present study, cipher uses the two quasigroups selected randomly thereby increasing the strength of the cipher as it would be difficult to search for a line of attack if the two quasigroup are secret as well as random. Same approach has been used by them [7] where every user can have their own pair of quasigroup of order 128 but stored in some public directory.

Generally working with quasigroups of large order is computationally infeasible (both in terms of space and in terms of time) to store the entire quasigroup and perform multiplication by a lookup table. This means there is a constant need for efficient method for representing quasigroups of large order.

The fast way of generating a quasigroup is to use the first row in the multiplicative table of the quasigroup [8]. The first row is in fact a permutation of the elements $\mathbb{Z}_p^* = \mathbb{Z}_p\setminus\{0\} = \{1, 2,\ldots, p\text{-}1\}$ and by knowing only that permutation it is possible to define the entire quasigroup from that. To begin with, the first permutation has to be known which generates a unique quasigroup corresponding to that permutation. In case of efficient encryption of messages in ASCII code is concerned, a quasigroup such that $0\notin Q$ won't solve the purpose. The same concept has been used by Gligoroski *et al.* [14].

Kościelny[15]-[16] enumerates a method through which we can have many isotope of a primary quasigroup. But for that the entire table of operation of the quasigroup has to be known priory.

Improvements in the area of generators of Quasigroup Completion problem is been reported by Barták [17]. He found the completion of a partial latin square representing the multiplication table of a quasigroup. So there exists a possibility to trackback the quasigroup if it does not form the part of secret key i.e. once the preset/fixed quasigroup is tracked down then the cipher is broken.

Generally while working with quasigroups of large order, it is computationally infeasible (both in terms of space and in terms of time) to store the entire quasigroup and perform multiplication by a lookup table.

The aim of this paper is to enhance the security of the existing block ciphers based quasigroups, by introducing a method for construction of huge quasigroups randomly. Quasigroup of large order depends mainly on the cryptographic algorithm and its other memory requirements. This paper defines a block cipher that initially generates two quasigroups of order256, such that neither the entire multiplicative table be stored before the encryption or

Deepthi Haridas, S Ventakaraman and Geeta Varadan are with Advanced Data Processing Research Institute (ADRIN), Department of Space, Government of India, Secunderabad, Andhra Pradesh, India (e-mail: deepthi@adrin.res.in)

decryption starts nor the first permutation has to be stored in memory for quasigroup constructions. Existing ciphers based on quasigroups uses preset quasigroups, thus taking many bytes of storage per quasigroup. Whereas in this approach some algebraic operations allows quasigroups(of huge order) to be used without any requisite for storage.

After constructing two random quasigroups of order 256 they have been tested for differential cryptanalysis by calculating the xor-distribution table for concerned quasigroup. There are more than 10^{58000} quasigroups with cardinality 256. In this manner, ruling out possibility of brute force attack. Hassinen and Markoviski. [18] claims that knowing the structure of the quasigroup reduces brute force complexity to n!. Which is never possible with this modification of random quasigroup instead of using a fixed set of quasigroup stored somewhere.

The organization of the paper is following: in section 2 comprises of basic definitions, section 3 enumerates properties of quasigroup string transformation, section 4 construction of random quasigroup, 5 Quasigroup block cipher is explained and section 6 gives the results and some cryptographical strength aspects of this cipher.

II. BASIC DEFINITIONS

A. Definitions

*Let Q = {a_1, a_2, ...,a_n} be a finite set of n elements. A quasigroup (Q, *) is a groupoid satisfying the law*

$$(\forall u,\ v \in Q)(\exists !x,\ y \in Q)\ \ u*x=v\ \&\ y*u=v. \qquad (1)$$

Closely related structure of quasigroup is the so called structure of a Latin square:

B. Definitions

A Latin square L on a finite set Q of cardinality |Q|=n is called an n x n–matrix L_{nxn}, with elements from Q such that each row and column of the matrix is a permutation of Q.

There is a simple Lemma that connects quasigroups and Latin Squares:

Lemma 1. *To any finite quasigroup (Q, *) given by its multiplicative table it is associated a Latin Square L_{nxn}, consisting of the matrix formed by the body of the table, and each Latin square L_{nxn} on a set Q define a multiplication table of a quasigroup (Q,*).*

Given a quasigroup (Q, *) five new operations $*^{-1}$, ${}^{-1}*$, ${}^{-1}(*^{-1})$, $({}^{-1}*)^{-1}$, $*^{*}$ on the set Q can be derived by:

$$*^{-1}(x, y) = z \iff x\mathbf{*}z=y$$
$${}^{-1}*(x, y) = z \iff z*y=x$$
$${}^{-1}(*^{-1})\,(x, y) = z \iff *^{-1}(z, y) = x \iff z\mathbf{*x}=y$$
$$({}^{-1}*)^{-1}(x, y) = z \iff {}^{-1}*(x, z) = y \iff y*z=x$$
$$*^{*}(x, y) = z \iff y*x=z$$

TABLE I
MATHEMATICAL SYMBOLS

Symbol	Meaning
Q	Finite set(i.e. alphabet)
Q^+	Finite strings(i.e. the set of all non empty words)formed by elements of Q
(Q,*)	Quasigroup
|Q|	Order of quasigroup.
α	Element of Q^+, $\alpha=a_1a_2\ldots a_n \forall\ a_i\in Q$
e_{a*}	e-transformation of Q^+ based on the operation * with leader a, i.e. $e_{a*}:Q^+\to Q^+$ then If $\beta=e_{a*}(\alpha)= b_1b_2\ldots b_n$, such that $b_i=b_{i-1}* a_i$ where $b_0=a\forall\ i=1,2\ldots n$
$d_{a\backslash}$	d-transformation of Q^+ based on the operation \with leader a, i.e. $d_{a/}:Q^+\to Q^+$ then if $\gamma=d_{a\backslash}(\alpha)= c_1c_2\ldots c_n$, such that $c_i=a_{i-1}\backslash a_i$ where a0=a $\forall$ i=1,2…n
e_{a1}	e_{a1*1}i.e. e_{a1} with $*_1$ operation
d_{a1}	d_{a1*1}i.e.d_{a1} with $*_1$ operation
$*_1, *_2, \ldots, *_k$	Sequence of quasigroup transformations
$\backslash_1, \backslash_2, \ldots,\backslash_k$	Sequence of quasigroup transformations Such that $\backslash_i=(*_i)^{-1}$the corresponding parastrophe $\forall$ i=1,2…n
E_k	E-transformation of Q^+ such that $E_k=E_{a1a2\ldots ak}=e_{a1}\circ e_{a2}\circ\ldots\circ e_{ak}$
D_k	D-transformation of Q^+ such that $D_k=D_{a1a2\ldots ak}=d_{a1}\circ d_{a2}\circ\ldots\circ d_{ak}$
T_k	T-transformation of Q^+ such that $T_k=t_{a1a2\ldots ak}=t_{a1}\circ t_{a2}\circ\ldots\circ t_{ak}$, where $t_{ai}\in\{e_{ai},d_{ai}\}$
T_k^{-1}	T-inverse transformation of Q^+ such that the inverse transformation has reverse order of used indexes of leader a_i and corresponding parastrophe i.e. $T_k^{-1}=t_{ak\ldots a1}=t_{ak}'\circ t_{ak-1}'\circ\ldots\circ t_{a1}'$, where $t_{ai}\in\{e_{ai},d_{ai}\}$
(T,T^{-1})	The pair of functions for any transformation considered as a pair of encryption and decryption functions for the strings on an alphabet Q
ϕ (n)	Euler's ϕ-function.

The set *Par* (*) = {*, $*^{-1}$, ${}^{-1}*$, ${}^{-1}(*^{-1})$, $({}^{-1}*)^{-1}$, $*^{*}$} is said to be set of parastrophes of *. Then, for each g∈ *Par*(*), (Q, g) is a quasigroup too and *Par*(*) =*Par*(g). Usually, for multiplicatively denoted quasigroup (Q,*), instead of $*^{-1}$, ${}^{-1}*$ one writes \, / respectively, and calls them left and right parastrophes. Then

$$x*y=z \iff y=x\backslash z \iff x=z/y \qquad (2)$$

Then the algebra (Q, *, \, /) satisfies the identities

$$x \backslash (x* y) =y,\ \ (x* y)/y=x,\ \ x* (x\backslash y) =y,\ (x/y) * y=x. \qquad (3)$$

III. QUASIGROUP STRING TRANSFORMATION

In this section the methods of quasigroup string transformations in brief and address several theorems which are proved in [6],[12]-[13],[19] and properties which they satisfy.

Consider the notations given in Table I, used throughput this paper. The functions E_k, D_k and T_k have many interesting properties, and for the cryptographical purpose the most important ones are the following:

Let us suppose the operations * and \ defined on an alphabet Q with their properties according to (2) and (3), and let $\alpha=a_1a_2\ldots a_n\in Q^+$, $a_i\in Q$, $a= a_0=b_0\in Q$. Then the transformation function e_{a*} based on the operations * and $d_{a\backslash}$ on \ are defined as in TableI as follows:

$\beta = e_{a*}(\alpha) = b_1 b_2 \ldots b_n$ and $\gamma = d_{a\backslash}(\alpha) = c_1 c_2 \ldots c_n$ then we have

$$b_i = b_{i-1} * a_i, \quad c_i = a_{i-1} \backslash a_i, \ \forall \quad i = 1, 2, \ldots, n \qquad (4)$$

Then the following is true:

Lemma2. [12] *For each $\alpha \in Q^+$, $e_{a*}(d_{a\backslash}\ (\alpha)) = \alpha = d_{a\backslash}\ (e_{a*}(\alpha))$.*

Lemma3. [12]*The transformation functions e_{a*}, $d_{a\backslash}$ are permutations on Q^+ and the inverse of e_{a*} is $(e_{a*})^{-1} = d_{a\backslash}$.*

Lemma4. [7]*The transformation functions E_k, D_k and T_k are permutations on Q^+.*

Other very important property from the security point of view is considered by the following theorem:

Theorem 1. *Consider an arbitrary string $\alpha = a_1 a_2 \ldots a_n \in Q^+$, where $a_i \in Q$, let $\beta = E_k(\alpha)$, $\gamma = D_k(\alpha)$.*

(a) [13] *If n is enough large then, for each l: $1 \leq l \leq k$, the distribution of substrings of β of length l is uniform.*

(b) [11] *If n and k are enough large, then the distribution of substrings of γ of a fixed length l ($l \geq 1$) is uniform.*

Theorem 2. *There are at least n! (n-1)!...2! 1! different Latin squares (quasigroups) on a set with cardinality n.*

From theorem 2 we can compute the lower bound for the number of 256*256. In fact from [20] the lower bound for the number of 256*256 Latin squares is 10^{58000}.

IV. Quick Construction of Random Latin Squares

Since the quasigroup(Q,*) is non-associative and non-commutative, the composition of e-transformations are fixed and it cannot be changed (this is not the case if the quasigroups are commutative or associative) i.e. there one-wayness is lost once the quasigroups attains commutative and associative properties. Ideally for cryptographic purposes quasigroup should be shapeless quasigroup [10].

C. Definition

A quasigroup ($Q, \circ$) of order n is said to be shapeless if it is non-commutative, non-associative, it does not have either left or right unit, it does not contain proper sub-quasigroups, and there is no $k < 2n$ for which are satisfied the identities of the kinds:

$$\underbrace{x^{\circ}(\ldots(x^{\circ}}_{k}y)) = y \text{ and } y = ((y^{\circ}\underbrace{x)^{\circ}\ldots)^{\circ}x}_{k} \qquad (5)$$

Quaisgroups of order 256 have been used in this paper which implies the memory amount needed to define one quasigroup is 64KB. Therefore if each time a new quasigroup has to be used, this concludes to an important cryptographical problem i.e. *generation of huge order quasigroup that could generate quasigroups with a very large number of elements without necessity of their storage.*

D. Definition

Let (Q_1,o) and (Q_2,) be two quasigroups with $|Q_1| = |Q_2|$. The ordered triplet (π_x, π_y, π_t) of one to one mappings π_x, π_y, π_t of the set Q_1 on to set Q_2 is called isotopism of Q_1 upon Q_2 if*

$$\pi_x(x) * \pi_y(y) = \pi_t(x o y) \quad \forall x, y \in G. \qquad (6)$$

In other words Q_2 is an isotope of a primary quasigroup Q_1. The set of all isotopisms of a quasigroup of order n forms a group of order $(n!)^3$.

To be noted that the mapping π_t permutes the elements in the table of operations in a quasigroup Q_1. While π_x and π_y operate on the elements of the row and column borders of this table. In short the Cayley table of quasigroup (Q_2,*) can be obtained from the Cayley table of quasigroups(Q_1,o).

From (6) it follows that

$$X * Y = \pi_t(\ \pi_x^{-1}(X) \text{ o } \pi_y^{-1}(Y)\ ) \qquad (7)$$

Usually isotopies are used to create a quasigroup, the only information that needs to be stored are the permutations π_x, π_y, π_t along with the group that is used to generate the quasigroup(Q_2,*).

D. Definition (Euler's ϕ-function): The function ϕ:N→N defined by the relations $\phi(1) = 1$ and $\phi(n)$ =number of positive integers less than n and relatively prime to n, for n>1, is known as Euler's ϕ-function.

The present study gives a method to construct the latin sqaure defined on the set Q= {0, 1, 2, …, 255} rather than for Q= {1, 2, …,p-1}, where prime 'p' is the order of quasigroup which are given in [8], [14]. If $0 \notin Q$ then the quasigroup construction won't be of use for encryption and decryption as the set considered is the ASCII code. The permutations $f_j(x)$ and $g_j(y)$ where $j \in \phi(n)$, operates on the elements of the row and column borders of the table. Hence the table can be obtained by applying (5) to $f_{\phi(n)}(x)$ and $g_{\phi(n)}(y)$.

Varying the value of k gives different quasigroup. As this cipher needs two sets of quasigroup, two sets of different f_k and g_k functions where $j \in \phi(n)$ results in random quasigroup construction. If $j \notin \phi(n)$, then j is selected as the closest possible element of $\phi(n)$. Since the cipher needs two quasigroups each time two random quasigroups would be generated depending on the k selected.

V. Quasigroup Block Cipher

The quasigroup block cipher makes use of two different quasigroups let it be (Q_1, $*_1$) and (Q_2, $*_2$) such that they are not mutually dual quasigroups. The corresponding quasigroup cipher consists of (Q_1, Q_2, $*_1$, $*_2$, $\backslash_1$, $\backslash_2$, e_a, d_a) all the symbols have been defined in Table I.

To encrypt a block $m_1 m_2 \ldots m_l$ of block length l with password $a_1 a_2 \ldots a_k$ of password length k. The encryption is done as follows:

$$(c_1c_2\ldots c_l)_i = \begin{cases} e_{a_i,*_1}(m_1,m_2,\ldots m_l), \\ \qquad\qquad \text{if } \mathrm{mod}(a_i,2)=0 \\ d_{a_i,*_2}(m_1,m_2,\ldots m_l), otherwise \end{cases} \quad (8)$$

$$\text{where } \; i = 1,2,\ldots,k$$

This works iteratively for k rounds such that the resulting ciphertext block $c_1c_2\ldots c_l$ for each iteration behaves as

$$(m_1m_2\ldots m_l) = (c_1c_2\ldots c_l)_{i-1}, \; \forall \; 1<i\leq k \quad (9)$$

For decryption of the ciphertext block $c_1c_2\ldots c_l$ of block length l with password $a_1a_2\ldots a_k$ of password length k. The order of using the password is reversed, the password for decryption is $a_ka_{k-1}\ldots a_1$ i.e. similar to (8) where for a[i] is the i^{th} position of $a_1a_2\ldots a_k$ for decryption the password is traversed in the reverse direction such that the new password for decryption is $a_1a_2\ldots a_k=a_ka_{k-1}\ldots a_1$ for encryption. The iteration is as follows:

$$(m_1m_2\ldots m_l)_i = \begin{cases} d_{a_i,\backslash_1}(c_1,c_2,\ldots c_l), \\ \qquad\qquad \text{if } \mathrm{mod}(a_i,2)=0 \\ e_{a_i,\backslash_2}(c_1,c_2,\ldots c_l), otherwise \end{cases} \quad (10)$$

$$\text{where } \; i = 1,2,\ldots,k$$

The quasigroup operations $\backslash_1$, $\backslash_2$ are the corresponding parastrophe of $*_1$ and $*_2$. Iteratively repeat (10) for k rounds, the corresponding relation (9) in this case is as follows:

$$(c_1c_2\ldots c_l)= (m_1m_2\ldots m_l)_{i-1}, \; \forall \; 1<i\leq k \quad (11)$$

VI. Results

The resistance of the method on statistical attacks seems to be very good. This claim is based on the facts from the point of view of differential cryptanalysis [18]. The tables were generated, of calculated differentials between xor-ed plain and cipher text, based on the two quasigroups which have been constructed randomly. It was observed that in every round the probability of transition from one xor-ed value of plain text to some other xor-ed value of ciphertext tend to be uniform distribution thus making an attack by differential cryptanalysis inefficient.

Next property to be observed was the distribution of characters of a plaintext and its ciphertext. Figure 1 illustrates the distribution of character of plaintext and of its ciphertext obtained by using different quasigroups generated from varying k over $\phi(n)$.

It can be observed from Figure1 that the distribution of letters for the ciphertexts obtained by using random quasigroup corresponding to the different values of k follows the same pattern. The ciphertexts obtained from different quasigroups operations are behaving in a similar way as to randomize the data. Figure 1 the input data is a raw data and the ciphertext is constructed out of the same raw data with same password ‘12’ and block length ‘2’.

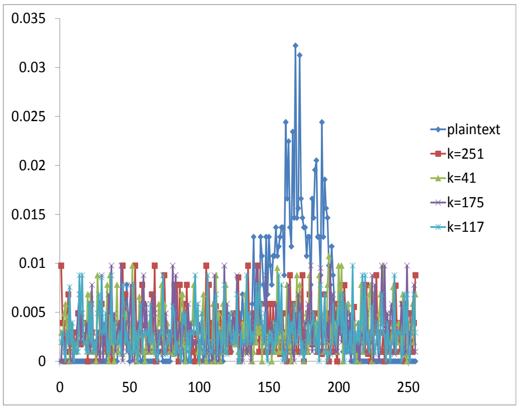

Fig. 1 Distribution of characters of the input raw data and its corresponding different ciphertexts.

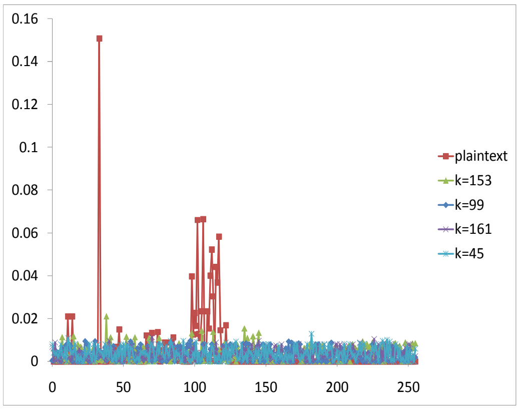

Fig. 2 Distribution of characters of the plaintext comprising text data and its corresponding ciphertexts generated.

Fig. 2 illustrates a graph comparing the distribution of characters of the plaintext as well as the generated set from different values of k. the plaintext comprises of a text file, all the ciphertext data are generated using the same password ‘12’ and block length ‘2’. The experiment had been repeated with almost all the quasigroups generated from by varying k, similar results were obtained i.e. the cipher with random quasigroup when applied to a string which is not completely random gives a randomized data.

References

[1] A. J. Menezes, P.C. Van Oorschot, S.A. Vanstone, Handbook of Applied Cryptography, CRC Press 1997

[2] J. Dénes and A. D. Keedwell, Latin Squares. New Developments in the Theory and Applications, North-Holland Publishing Co., Amsterdam, 1981.

[3] J. Dénes and A. D. Keedwell: A New Authentication Scheme based on Latin Squares, Discrete Mathematics, no. 106/107, (1992), pp. 157-162.

[4] Hall, M.: Combinatorial theory, Blaisdell Publishing Company, 1967.

[5] Bakhtiari, S., Safavi-Naini, R., Pieprzyk, J., “A Message Authentication Code Based on Latin Squares, Proc. Australasian Conference on Information Security and Privacy”, pp. 194–203, 1997.

[6] S. Markovski, D. Gligoroski, S. Andova, Using quasigroups for one-one secure encoding, Proc. VIII Conf. Logic and Computer Science “LIRA ’97”, Novi Sad, pp. 157–162, 1997.

[7] S. Markovski, D. Gligoroski, B. Stojčevska, Secure two-way on-line communication by using quasigroup enciphering with almost public key, Novi Sad Journal of Mathematics, vol. 30, 2000.

[8] S.I. Marnas, L. Angelis and G.L. Bleris, All-Or-Nothing Transform Using Quasigroups, Proc. 1st Balkan Conference in Informatics, pp. 183-191, 2003.

[9] V. A. Shcherbacov, On linear and inverse quasigroups and its applications in code theory, 2003, www.karlin.mff.cuni.cz/drapal/speccurs.pdf

[10] D. Gligoroski, S. Markovski, L. Kocarev, EdonR, An Infinite Family of Cryptographic Hash functions, International Journal of Network Security, Vol. 8(3), pp. 293-300, 2009

[11] D. Gligoroski, Edon - library of reconfigurable cryptographic primitives suitable for embedded systems, Workshop on cryptographic hardware and embedded systems, 2003.

[12] S. Markovski, D. Gligoroski, V. Bakeva: Quasigroup String Processing: Part 1, Maced. Acad. of Sci. and Arts, Sc. Math. Tech. Scien. XX 1-2, pp. 13–28, 1999.

[13] S. Markovski, V. Kusakatov: Quasigroup String Processing: Part 2, Contributions, Sec. math. Tech.Sci., MANU, XXI, vol. 1-2, pp. 15–32 2000.

[14] D. Gligoroski, Stream Cipher based on Quasigroup string transformations in $\mathbb{Z}_p^*$, arXiv:cs.CR/0403043, Macedonian Academy of Science and Arts, annual Proceedings in Mathematical and Technical Sciences, 2004.

[15] Kościelny, C.: Generating Quasigroups for cryptographic applications, Int. J. Appl. Math. Sci., Vol. 12, No.4, pp. 559-569, 2002.

[16] Kościelny, C.: A method of constructing quasigroup-based stream-ciphers. Appl. Math. and Comp. Sci. vol 6,pp. 109–121, 1996.

[17] Roman Barták, On generators of Random Quasigroup Problems, In proc. Of ERCIM 05 workshop, pp. 264-278, 2006.

[18] M Hassinen, S Markoviski, Differential Cryptanalysis of the quasigroup cipher, Proceedings of the Finnish Data Processing Week, Petrozavodsk, Russia. Petrozavodsk State University, (2004).

[19] D. Gligoroski, S. Markovski, Cryptographic potentials of quasigroup transformations, Talk at EIDMA Cryptography Working Group, Utrecht, 2003.

[20] McKay, B.D., Rogoyski, E.:Latin squares of order 10, Electronic J. Comb.2,1995, http://ejc.math.gatech.edu:8080/Journal/journalhome.html