MULTIPERMUTATIONS IN CRYPTO WORLD: DIFFERENT FACES OF THE PERFECT DIFFUSION LAYER

ALEKSANDRA MILEVA

Abstract. Diffusion layers, and specially perfect diffusion layers, are very important subject for cryptographic research. Main quest is a perfect diffusion layer with more optimal hardware and/or software implementations (if possible, the last needs to holds also for its inverse). Different structures can be used for representing these layers, but all are interconnected. We start with multipermutations as a tools for obtaining perfect diffusion, and we summarize the interconnections between them, MDS codes, Latin squares and quasigroups, orthogonal arrays and m-arcs. We give a new construction of perfect recursive diffusion layer from r-recursive MDS codes, or recursively r-differentiable quasigroups.

1. Introduction

The concept of diffusion was first introduced by Shannon in its seminal paper "Communication Theory of Secrecy Systems" [34], and refers to dissipating the statistical structure of plaintext into statistical structure involving long combinations of letters in ciphertext. Modern block ciphers are cascades of rounds, and every round has a diffusion layer as its building block, usually some linear transformation. For a long time diffusion layers were little bit neglected in the cryptographic research, compared to confusion layers. More serous research started with introduction of notation of multipermutation by Schnorr and Vaudenay [30, 31]. By their definition, a permutation $f : Z^2 \to Z^2$, $f(x, y) = (f_1(x, y), f_2(x, y))$ is a **multipermutation** if for every $x, y \in Z$ the mappings $f_i(a, *), f_i(*, b)$ for $i = 1, 2$ are permutations on Z. Vaudenay [37] generalized the concept of multipermutation with $(r, t)-$multipermutation and introduced the notation of perfect diffusion in the sense that changing i input values changes at least $(t - i + 1)$ output values.

Definition 1.1. ([37]) A function $f : Z^r \to Z^t$ is a $(r, t)-$**multipermutation** if any two different $(r+t)$-tuples of the form $(x_1, \ldots, x_r, f(x_1, \ldots, x_r))$ cannot collide in any r positions.

The perfect diffusion with $(r, t)-$ multipermutation is achieved because for any $\mathrm{x}_1, \ldots, \mathrm{x}_\mathrm{r} \in Z$ and any integer i such that $1 \leq i \leq r$, changing i input values on $f(\mathrm{x}_1, \ldots, \mathrm{x}_\mathrm{r})$, will change at least $(t - i + 1)$ output values. Linear transformations with maximal branch numbers (introduced by Daemen [10]) have perfect diffusion. Perfect diffusion provides resistance against linear and differential cryptanalysis, so building perfect diffusion layers, that have, together with their inverses, optimal performances in hardware and/or software, is one of the main goals in modern cryptography.

2000 *Mathematics Subject Classification.* Primary , Secondary .

Key words and phrases. Multipermutations, MDS codes, n-ary quasigroups, Latin squares.

In this paper, we try to summarized some known and some unknown interconnections between multipermutations, MDS codes, Latin squares and quasigroups, orthogonal arrays and m-arcs. We also give short survey of recent prospects in building perfect diffusion layers, and a new construction of perfect recursive diffusion layer from r-recursive MDS codes, or recursively r-differentiable quasigroups.

2. Connection with MDS codes

Maximum Distance Separable (MDS) code is a linear $[n, k, d]$ code that meets the Singleton bound, i.e. $d = n - k + 1$, where n is the length of the codeword, k is the dimension of code, and d is the minimal distance. Known MDS codes are Reed Solomon Codes, $(3, 1)$ Hamming code, $(4, 1)$ extended Hamming codes, dual code of MDS code, etc. The following two theorems hold for MDS codes.

Theorem 2.1 ([26], page 319)**.** Let C be a $[n, k, d]$ code over finite field $\mathbb{F}_q$. The following properties are equivalent:

- C is MDS.
- Every k columns of a generator matrix are linearly independent.
- Every $n - k$ columns of a parity check matrix are linearly independent.
- C has minimum weight codewords in any d coordinates.

Theorem 2.2 ([26], page 321)**.** A $[n, k, d]$ code C with generator matrix $[I|A]$, where A is a $k \times (n - k)$ matrix, is MDS if and only if every square submatrix formed from any i rows and any i columns (for any $i \in \{1, \ldots, min(k, n - k)\}$) of A is nonsingular.

From the definition, the set of all words of the form $(\mathrm{x}_1, \ldots, \mathrm{x}_\mathrm{r}, f(\mathrm{x}_1, \ldots, \mathrm{x}_\mathrm{r}))$ can be seen as a systematic error correcting code of $|Z^r|$ words of length $r + t$ with minimal distance $t + 1$, which matches the Singleton bound. The connection between multipermutations and MDS codes can be seen from the fact that if, Z is a finite field, a linear (r, t)-multipermutation is a $[r + t, r, t + 1]$ MDS code, in a sense that any word of length r is coded by the concatenation of the word and its multipermutation image. Equivalence holds only when MDS code minimal distance is at least r $(t+1 \geq r)$. Linear $(r, t)-$ multipermutation f can be represented using the $r \times t$ MDS matrix M as $f : x \mapsto M \times x$. The following two propositions give the characterization of MDS matrices over a finite field $\mathbb{F}_q$.

Proposition 2.3. *([26]) A square matrix M with coefficients in $\mathbb{F}_q$ is an MDS matrix if and only if every square submatrices of M are nonsingular.*

Proposition 2.4. *([22]) All entries of an MDS matrix with coefficients in $\mathbb{F}_q$ and its inverse are non zero.*

Many block ciphers, as Shark [28], Square[12], AES [11], Twofish [33] etc, use MDS matrices for obtaining perfect diffusion. Usually, MDS matrices are over finite fields $GF(2^8)$ or $GF(2^4)$. Block ciphers (including lightweight BCs) usually have block sizes of $32, 64$ or 128 bits, so usually they use

- $(4, 4)$-multipermutations, represented by 4×4 MDS matrices ($[8, 4, 5]$ MDS codes) of $w = 8, 16$ and 32-bit words, or
- $(8, 8)$-multipermutations, represented by 8×8 MDS matrices ($[16, 8, 9]$ MDS codes) of $w = 4, 8$ and 16-bit words.

Most of the used MDS matrices are circulant or Hadamard. For example, AES (Rijndael) [11] in its famous *MixColumn* operation, uses a 4×4 MDS circulant matrix $M = cir(0x02, 0x03, 0x01, 0x01)$ (as $(4,4)$-multipermutation) with coefficients in $GF(2^8)$ (first used in Square [12]), which is represented as $GF(2)[x]/v(x)$ where $v(x) = x^8 + x^4 + x^3 + x + 1$ is a irreducible polynomial of degree 8 over $GF(2)$. Circulant MDS matrices are used also in Whirlpool [4] and Maelstrom-0 [16] ($M = cir(0x01, 0x01, 0x04, 0x01, 0x08, 0x05, 0x02, 0x09)$ as $(8,8)$-multipermutation), Gr$\oslash$stl [19] ($M = cir(0x02, 0x02, 0x03, 0x04, 0x05, 0x03, 0x05, 0x07)$ as $(8,8)$-multipermutation), etc.

Let H be a $m \times m$ Hadamard matrix $H = had(a_0, a_1, \ldots, a_{m-1})$ with elements $h_{i,j} = a_{i \oplus j}$. Each Hadamard matrix H over a finite field has the following properties: $H^2 = \gamma \cdot I$ where γ is a constant. When $\gamma = 1$, H is an involution matrix. This property provides the same matrix to be used for encryption and decryption. Involutional Hadamard MDS matrices can be found in block ciphers KHAZAD [3] (with linear diffusion layer θ such that $\theta(a) = b \Leftrightarrow b = a \cdot H$ and $H = had(0x01, 0x03, 0x04, 0x05, 0x06, 0x08, 0x0B, 0x07)$ as $(8,8)$-multipermutation), Anubis [2] ($H = had(0x01, 0x02, 0x04, 0x06)$ as $(4,4)$-multipermutation), CLEFIA [35] (two $(4,4)$-multipermutations obteined by two 4×4 Hadamard matrices), etc.

2.1. **Building Efficient MDS Matrices in Software.** Junod and Vaudenay [24] give a construction of 4×4 and 8×8 MDS matrices efficient in software, by considering software implementations on various platforms. They considered only one direction, so obtained inverse matrices may not be very efficient in software. This is not important if MDS matrix is used with constructions like Feistel or Lai-Massey schemes which do not use inverse matrix in decryption.

Their strategy is to maximize the number of 1s in the matrix and to minimize the number of different constants. In this way, the "optimal" 4×4 MDS matrix has 9 ones and 3 different values and the "optimal" 8×8 MDS matrix has 24 ones and 5 different values. One example of "optimal" 4×4 MDS matrix can be found in FOX64 block cipher [25], which uses Lai-Massey scheme with an orthomorphism and F function as a S-P network, and it is given by

$$M = \begin{pmatrix} 0x01 & 0x01 & 0x01 & 0x02 \\ 0x01 & 0xFD & 0x02 & 0x01 \\ 0xFD & 0x02 & 0x01 & 0x01 \\ 0x02 & 0x01 & 0xFD & 0x01 \end{pmatrix}$$

2.2. **Building Efficient MDS Matrices in Hardware.** Guo, Peyrin, and Poschmann [20] give a construction of the dxd MDS matrices for compact hardware implementation (this holds for inverse also) - PHOTON family of lightweight hash functions.

First they take a matrix A that updates the last cell of the column vector with a linear combination of all of the vector cells and then rotates the vector by one

position towards the top. A is $d \times d$ matrix of the form

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \\ 0 & 0 & 0 & 0 & \cdots & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 1 \\ Z_0 & Z_1 & Z_2 & Z_3 & \ldots & Z_{d-4} & Z_{d-3} & Z_{d-2} & Z_{d-1} \end{pmatrix}$$

where coefficients $(Z_0, Z_1, \ldots, Z_{d-1})$ can be chosen freely. Such a matrix is denoted by $Serial(Z_0, Z_1, \ldots, Z_{d-1})$. Then, their MDS matrix M is obtained by d applications of matrix A, or $M = A^d$. For different size d, the authors tested all the possible values of $(Z_0, Z_1, \ldots, Z_{d-1})$, and picked the most compact candidate making A^d an MDS matrix. In this paper, the authors give $A = Serial(1, 2, 1, 4)$ as an example of 4×4 MDS matrix. LED lightweight block cipher [21] uses $M = A^4 = Serial(4, 1, 2, 2)^4$ as MDS matrix.

The efficiency of this approach lies in the shifting property of A, since this allows to re-use the existing memory with neither temporary storage nor additional control logic required. This new diffusion layer does not affect very much software implementations, because precomputed lookup tables can be used here also.

Gupta and Ray [22] further investigate $M = A^d$ matrices for better hardware implementation. They try to maximize the occurrence of 1s in the coefficients. There are no 4×4 MDS matrices of this type with three or four 1s. If $Z_0 = 1$, the hardware footprint for decryption is as good as that of encryption circuitry. They give the following characterization of $A = Serial(Z_0, Z_1, Z_2, Z_3)$ for A^4 to be an MDS matrix, with restriction of the Z_i values to the set $\{1, \alpha, \alpha^2, \alpha + 1\}$, where α is the root of constructing polynomial of the field $\mathbb{F}_{2^n}$, and number of 1's is maximized:

- $A = Serial(1, \alpha, 1, \alpha^2)$, for all $n \geq 5$ except when $n = 6$ and α is root of $x^6 + x^5 + x^4 + x + 1$.
- $A = Serial(1, \alpha, 1, \alpha + 1)$, for all $n \geq 4$
- $A = Serial(\alpha, 1, 1, \alpha^2)$, for all $n \geq 5$ except when $n = 4$ and α is root of $x^4 + x^3 + x^2 + x + 1$ or $x^4 + x^3 + 1$ or when $n = 7$ and α is root of $x^7 + x^6 + x^5 + x^4 + 1$.

Similarly, generic constructions of 5×5 MDS matrices of the form $Serial(1, \alpha, 1, 1, \alpha^2)^5$ and $Serial(1, \alpha^2, 1, 1, \alpha)^5$ are proposed, when some conditions are fulfilled. Authors also give the conditions when A of the form $Serial(1, \beta, 1, \beta^2)$, $Serial(1, \beta, 1, \beta + 1)$ and $Serial(\beta, 1, 1, \beta^2)$, defined for any non zero $\beta \in \mathbb{F}_{2^n}$, generates a 4×4 MDS matrix A^4. Additionally, if $Serial(1, \beta, 1, \beta^2)^4$ is an MDS matrix, then so is the matrix $Serial(1, \beta, 1, \beta^2)^{-4}$.

Sajadieh et al [29] define the notion of a recursive diffusion layer and propose a method to construct such perfect diffusion layers.

Definition 2.5. A diffusion layer D with r words x_i as the input, and r words y_i as the output is called a **recursive diffusion layer** if it can be represented in the

following form:

$$D : \begin{cases} y_0 = x_0 \oplus F_0(x_1, \ldots, x_{r-1}) \\ y_1 = x_1 \oplus F_1(x_2, \ldots, x_{r-1}, y_0) \\ \vdots \\ y_{r-1} = x_{r-1} \oplus F_{r-1}(y_0, \ldots, y_{r-2}) \end{cases}$$

where $F_0, F_1, \ldots, F_{r-1}$ are arbitrary functions.

Sajadieh et al [29] considered the vector space $\mathbb{F}_2^n$ over $\mathbb{F}_2$, because multiplications with elements in $\mathbb{F}_{2^n}$ are specific linear transformations of vector space $\mathbb{F}_2^n$. For every linear transformation L of $\mathbb{F}_2^n$, there is a square matrix M over $\mathbb{F}_2$ such that $L(\mathbf{v}) = M \cdot \mathbf{v}$. So, the invertibility of L is equivalent to the non-singularity of M. They chose $L_i = \sum_{j=-1}^{2} a^{(j)} \cdot L^j$, where $a^{(j)} \in \mathbb{F}_2$ and $1 \leq i \leq r$. The final matrix obtained from this strategy can be treated as an $rn \times rn$ matrix over $\mathbb{F}_2$ or an $r \times r$ matrix composed of linear transformations over $\mathbb{F}_2^n$. In this way, classical field multiplications are replaced by simple $\mathbb{F}_2$ linear transformations (combinations of XORs and shifts) which are much lighter. The authors give a construction of one class of perfect recursive diffusion layers from same $F_i(x_1, x_2, x_3) = x_2 \oplus x_3 \oplus L(x_1 \oplus x_3)$ by the following theorem.

Theorem 2.6. ([29]) For the diffusion layer represented in

$$D : \begin{cases} y_0 = x_0 \oplus x_2 \oplus x_3 \oplus L(x_1 \oplus x_3) \\ y_1 = x_1 \oplus x_3 \oplus y_0 \oplus L(x_2 \oplus y_0) \\ y_2 = x_2 \oplus y_0 \oplus y_1 \oplus L(x_3 \oplus y_1) \\ y_3 = x_3 \oplus y_1 \oplus y_2 \oplus L(y_0 \oplus y_2) \end{cases}$$

if the four linear functions $L(x), x \oplus L(x), x \oplus L^3(x)$, and $x \oplus L^7(x)$ are invertible, then this diffusion layer is perfect.

Example of 4×4 MDS matrix $A = Serial(1, 2, 1, 4)$ in [20] can be represented by recursive diffusion layer D with $F_i(x_1, x_2, x_3) = L(x_1) \oplus x_2 \oplus L^2(x_3)$, where $L(x) = 2x$ and $x \in GF(2^4)$.

MDS codes exist also over algebraic structures other than fields, like modules [15], commutative rings [7], finite principal ideal rings [14], cyclic groups [39], etc. This fact offers many new possibilities to construct efficient and compact hardware implementation of MDS matrices.

Wu, Wang and Wu [38] revisit the design strategy of [20] and [29], and construct recursive diffusion layers over particular commutative ring, with one or several bundle-based LFSRs, which have smaller hardware implementation.

Let L be an $n \times n$ non-singular matrix over $\mathbb{F}_2$ and

$$S = \{\sum a_{-i} L^{-i} + a_0 + \sum a_j L^j : i, j \in \mathbb{Z}^+, a_{-i}, a_0, a_j \in \mathbb{F}_2\}$$

be a set which includes all polynomials of L and L^{-1}. Then, the set S together with the addition of $\mathbb{F}_2$ and the multiplication of polynomials, form a commutative ring. It is denoted by $\mathbb{F}_2[L, L^{-1}]$. The following construction strategy is used in [38]:

- Construct $r \times r$ matrix $A = (A_{i,j})$ with each $A_{i,j} = \sum a_k^{i,j} \cdot L^k \in \mathbb{F}_2[L, L^{-1}]$.
- Choose an integer d and compute $D = A^d$ $(d \geq 1)$ as the final diffusion layer. D is perfect if and only if the determinant of each square submatrix of D is an invertible element in $\mathbb{F}_2[L, L^{-1}]$.

- Additionally, when choosing, $A_{i,j}$ should be with few terms and linear transformation L should be with no more than one XOR gate.

Their proposals have smaller hardware implementations than diffusion layers given in PHOTON lightweight hash family [20] and [29], because they use the smallest possible number of XOR gates. For $r = 4$ the best result is $A^{(4)}_{lfsr} = [L, 1, 1, L^2]$ and authors suggest also several lightweight linear transformations L. With several bundle-based LFSRs (Type-II Generalized Feistel Structure), for $r = 4$ the best result is $A^{(4)}_{gfs} = [L, 1, 1, L]$ and for $r = 8$ the best result is $A^{(8)}_{gfs} = [1, L^4, 1, L^{-1}, 1, L, 1, L^2]$. The authors discuss also some possible manners to construct perfect diffusion layers by increasing the number of iterations and using bit-level LFSRs.

Augot and Finiasz [1] try to construct larger MDS matrices, using the same previous recursive construction: 8×8 diffusion matrices with symbols of 4 bits and 16×16 diffusion matrices with symbols of 8 bits.

Berger [6] give a generic construction of MDS recursive diffusion layers from Gabidulin codes [18], which have additional MRD (Maximum Rank Distance) property. This fact gives an additional property to diffusion layers which seems interesting for cryptographic applications.

Recently, a new construction of block-wise MDS matrices over a finite commutative ring with identity is given by Dehnavi et al [13].

3. Connection with Latin Squares and Quasigroups

3.1. **Mathematical preliminaries.** A *quasigroup* is a groupoid $(Q, *)$ that satisfies the property for each one of the equations $a * x = b$ and $y * a = b$ to have a unique solution x, respectively y. When Q is a finite set, the main body of the Cayley table of the quasigroup $(Q, *)$ represents a Latin square, i.e., a matrix with rows and columns that are permutations of Q. Two quasigroups $(Q, *_1)$ and $(Q, *_2)$ are *orthogonal* if for any u and v in Q, there exists a unique pair of elements x and y of Q such that $x *_1 y = u$ and $x *_2 y = v$. In particular, if $(Q, *_1)$ and $(Q, *_2)$ are orthogonal and x and y run through all elements of Q, the ordered pairs $(x *_1 y, x *_2 y)$ run through all elements of Q^2. Moreover a set $\{(Q, *_i) \ | i = 1 \dots t, \ t \geq 2\}$ of quasigroups of order n is said to be orthogonal if any two distinct quasigroups are orthogonal. Such a set of pairwise orthogonal quasigroups is said to be a set of *mutually orthogonal quasigroups* or, when we speak about Latin squares - a set of *mutually orthogonal Latin squares* (*MOLS*). The maximal possible number of MOLS of order n is $(n - 1)$ and if so, the set is said to be *complete*.

An n-ary groupoid ($n \geq 1$) is an algebra (Q, h) on a nonempty set Q as its universe and with one n-ary operation $h : Q^n \to Q$. We use the definition of an n-ary quasigroup from Belousov [5].

Definition 3.1. An n-ary groupoid (Q, h) is an **n-ary quasigroup** (of order $|Q|$) if any n elements of the $a_1, a_2, \dots, a_{n+1} \in Q$, satisfying the equality

$$h(a_1, a_2, \dots, a_n) = a_{n+1},$$

uniquely specifies the remaining one.

2-ary quasigroups, 3-ary quasigroups and 4-ary quasigroups are also known as binary, ternary and quaternary quasigroups, respectively. When we say a quasigroup, we mean of a binary quasigroup.

To every finite n-ary quasigroup of order r, an equivalent combinatorial structure n-dimensional Latin hypercubes of order r can be associated. Let Q be the set of r different elements. By *n-dimensional Latin hypercube H of order r* we mean an n-dimensional array of r^n cells, where the cell contains an element of Q and where every set of r cells which coordinates match between themselves except in one coordinate, contains each of the elements of Q. Latin hypercubes of dimension 1, 2 and 3 are commonly called permutations, Latin squares, and Latin cubes, respectively.

3.2. **Connections.** A $(1,t)$-multipermutation is equivalent to a vector of t permutations. A $(2,1)$-multipermutation is equivalent to a Latin square (quasigroup), every $(2,2)$-multipermutation is equivalent to a pair of MOLS (mutually orthogonal quasigroups) and a $(2,t)$-multipermutation is equivalent to a set of t pairwise orthogonal Latin squares (quasigroups).

CS-Cipher [36] uses non-linear $(2,2)$-multipermutation M which maps two 8-bit words x_l and x_r, in the following way:

$$M(x_l, x_r) = (P(\varphi(x_l) \oplus x_r), P(R_l(x_l) \oplus x_r))$$

where P is a non-linear byte permutation, R_l is a bit rotation by one position to the left and φ is a linear byte permutation.

Let $Q = \{0,1\}^8$. We can rewrite function M, by two orthogonal quasigroups $(Q, *_1)$ and $(Q, *_2)$ of order 256 in the following way:

$$M(x_l, x_r) = (x_l *_1 x_r, x_l *_2 x_r)$$

where $x *_1 y = P(\varphi(x) \oplus y)$ and $x *_2 y = P(R_l(x) \oplus y)$.

Couselo et al [8] give a definition of k-recursive code and the conditions when this code is an MDS code.

Definition 3.2. Let $Q = \{a_1, a_2, \ldots, a_q\}$ be a finite set. A code C is called a **k-recursive code** if there exists a function $f : Q^k \to Q$ $(k \leq n)$ such that C is the set of all words $(u_0, \ldots, u_{n-1})$ satisfy the condition $u_{i+k} = f(u_i, \ldots, u_{i+k-1})$ for $i = 0, 1, \ldots, n-k-1$, where $u_0, \ldots, u_{n-1}$ are arbitrary elements of Q. It is denoted by $C(n, f)$.

$C(n, f)$ can be represented by

$$C(n,f) = \{(u_0, \ldots, u_{k-1}, f_0(u_0, \ldots, u_{k-1}), \ldots, f_{n-k-1}(u_0, \ldots, u_{k-1})) : (u_0, \ldots, u_{k-1}) \in Q^k\}$$

where
$f_0 = f_0(u_0, \ldots, u_{k-1}) = f(u_0, \ldots, u_{k-1})$,
$f_1 = f_1(u_0, \ldots, u_{k-1}) = f(u_1, \ldots, u_{k-1}, f_0)$
...
$f_{k-1} = f_k(u_0, \ldots, u_{k-1}) = f(u_{k-1}, f_0 \ldots, f_{k-2})$
$f_{i+k} = f_{i+k}(u_0, \ldots, u_{k-1}) = f(f_i, \ldots, f_{i+k-1})$ for $i \geq 0$
are **recursive derivatives**.

Theorem 3.3. ([8, 9]) The arbitrary k-recursive code $C(n, f)$ is an MDS code if and only if the following two conditions are fulfil:
(i) For every function f_i $(0 \leq i \leq n-k-1)$ holds that (Q, f_i) is k-ary quasigroup.
(ii) The system of functions $f_0, \ldots, f_{n-k-1}$ is an orthogonal system k-ary quasigroups, i.e. the values of arbitrary k functions from the set

$$(u_0, \ldots, u_{k-1}, f_0(u_0, \ldots, u_{k-1}), \ldots, f_{n-k-1}(u_0, \ldots, u_{k-1}))$$

uniquely specifies the remaining $n-k$ values.

Definition 3.4. ([8, 9]) A quasigroup operation f is called **recursively r-differentiable** if all its recursive derivatives $f_1, f_2, \ldots, f_r$ are quasigroup operations.

Definition 3.5. A quasigroup (Q, f) is called **recursively r-differentiable** if all recursive derivatives $f_1, f_2, \ldots, f_r$ of f are quasigroup operations.

Let $F_0 = F_1 = \ldots = F_{r-1} = F$ in the recursive diffusion layer D from [29]. Let $f(x_0, x_1, \ldots, x_{r-1}) = x_0 \oplus F(x_1, \ldots, x_{r-1})$. Then we can rewrite D as

$$D : \begin{cases} y_0 = x_0 \oplus F(x_1, \ldots, x_{r-1}) = f(x_0, x_1 \ldots, x_{r-1}) \\ y_1 = x_1 \oplus F(x_2, \ldots, x_{r-1}, y_0) = f(x_1, x_2, \ldots, x_{r-1}, y_0) \\ \vdots \\ y_{r-1} = x_{r-1} \oplus F(y_0, \ldots, y_{r-2}) = f(x_{r-1}, y_0, \ldots, y_{r-2}) \end{cases}$$

so we proofed the following Theorem.

Theorem 3.6. Every recursive diffusion layer D with r words x_i as the input, r words y_i as the output, and $F_0 = F_1 = \ldots = F_{r-1} = F$ is an $r-$recursive code $C(2r, f)$. If $C(2r, f)$ is MDS code, then D is a perfect diffusion layer.

Construction of a perfect recursive diffusion layer, besides using the $r-$recursive code, can be formulated using r-ary quasigroup by the following Theorem.

Theorem 3.7. Let (Q, f) be an r-ary recursively r-differentiable quasigroup. If the recursive derivatives $f_0, f_1, \ldots, f_{r-1}$ form an orthogonal system of r-ary quasigroups, then the diffusion layer D given by

$$D : \begin{cases} y_0 = f(x_0, x_1 \ldots, x_{r-1}) \\ y_1 = f(x_1, x_2, \ldots, x_{r-1}, y_0) \\ \vdots \\ y_{r-1} = f(x_{r-1}, y_0, \ldots, y_{r-2}) \end{cases}$$

is a perfect recursive diffusion layer.

Proof. Direct consequence from previous two Theorems.

Corollary 3.8. *Every recursive diffusion D layer with r words x_i as the input, r words y_i as the output, and $F_0 = F_1 = \ldots = F_{r-1} = F$ which represent MDS matrix can be represented using r applications of one $r-$ary quasigroup.*

Example of 4×4 MDS matrix $A = Serial(1, 2, 1, 4)$ in [20] can be represented by 4-recursive MDS code with the function

$$f(x_0, x_1, x_2, x_3) = x_0 \oplus L(x_1) \oplus x_2 \oplus L^2(x_3)$$

where $L(x) = 2x$ and $x \in GF(2^4)$. This means that $(GF(2^4), f)$ is $4-$ary quasigroup of order 16, and A can be obtained by 4 applications of the $4-$ary quasigroup operation f.

Couselo et al [8] give several constructions of linear MDS recursive codes, by using linear recurrence sequences (LSR) over module and finite field. Additionally, Markov and Nechaev [27] give a description of several recursive MDS codes by using LSR over finite field. Let $P = \mathbb{F}_q$. A **sequence over** P is a function $u : N_0 \to P$,

with representation $u = (u(0), u(1), \dots u(i), \dots)$. Let denote $P^{<1>} = \{u : N_0 \to P\}$. For an arbitrary monic polynomial

$$f(x) = x^m - f_{m-1}x^{m-1} - \dots - f_1x^1 - f_0 \in P[x]$$

with $L_P(f) = \{u \in P^{<1>} : u(i+m) = f_{m-1}u(i+m-1) + \dots + f_0u(i)\}$ is denoted the set of all LRS with characteristic polynomial $f(x)$. For any $n \geq m$ and any $u \in L_P(f)$ its initial segment of length n is $u(0, n-1) = (u(0), u(1), \dots u(i))$. The set:

$$C = L_P^{0,n-1}(f) = \{u(0, n-1) : u \in L_P(f)\}$$

is an $[n, m, d]_q-$code over P, called **linear recursive** $[n, m, d]-$**code with characteristic polynomial** $f(x)$.

Remark. ([27]) All the recursive $[8, 4, 5]_8-$MDS codes are BCH-codes.

4. Connection with with Orthogonal Arrays and m-arcs

4.1. **Connection with Orthogonal Arrays.** An **orthogonal array** $OA_\lambda(k, n, q)$ is $\lambda q^k \times n$ array of q symbols, such that in any k columns of the array every one of the possible q^k ordered k-tuples of symbols occurs in exact λ rows. Usually k is referred to as the *strength* of the OA, n is called the number of *factors (constrains)*, q is called the number of *levels* and λ is called the *index* of OA.

Theorem 4.1. ([23]) The rows of a $OA_1(k, n, q)$ linear orthogonal array of index unity and symbols from $\mathbb{F}_q$ are the codewords of a $[n, k, n-k+1]$ MDS code (linear $(k, n-k)-$multipermutation) over $\mathbb{F}_q$ and conversely.

4.2. **Connection with m-arcs.** Let $V[N+1, q]$ be a vector space. The lattice of subspaces of $V[N+1, q]$ of dimension at least 1 is called a Projective Geometry and is denoted by $PG(N, q)$. The 1-dimensional subspaces are called points, the 2-dimensional subspaces lines, the 3-dimensional subspaces planes, etc. of $PG(N, q)$. A set of m points in PG(N,q), with $m > N$, is called an m**-arc** if every $N+1$ of the points are linearly independent. Let K be a set of m points, $P_1, P_2, \dots, P_m$ in $PG(N, q)$. Form the $N+1 \times m$ matrix G whose m columns are projective coordinates of each of the points.

Theorem 4.2. (reformulated from [32]) K is an m-arc in $PG(N, q)$ if and only if G is the generator matrix of an $[m, N+1, m-N]$ q-ary MDS code.

In $PG(2, q)$ the largest m-arcs have size $q+1$ if q is odd and $q+2$ if q is even. A $(q+1)$-arc in $PG(2, q)$ is called an *oval* and a $(q+2)$- arc is called a *hyperoval*. In $PG(3, q)$, the largest m-arcs have size $q+1$. For every q there exists a $(q+1)$-arc in $PG(3, q)$, called a *twisted cubic*. Ovals give rise to $[q+1, 3, q-1]$ MDS codes $((3, q-2)-$multipermutations). Hyperovals give rise to $[q+2, 3, q]$ MDS codes $((3, q-1)-$multipermutations). Twisted cubics give rise to $[q+1, 4, q-2]$ MDS codes $((4, q-3)-$ multipermutations).

5. Future work

Last few years we are faced with intensive research on this subject, specially, different ways of their construction and finding perfect diffusion layers with optimal hardware and/or software implementation. Till recently, cryptographers used only MDS codes over fields for their construction, but diversity of MDS codes over different algebraic structures, offer new ways and new possibilities. For example,

recursive MDS codes by using linear recurrence sequences over module and BCH-codes are good candidates for further investigations.

References

[1] **D. Augot, M. Finiasz**: *Exhaustive Search for Small Dimension Recursive MDS Diffusion Layers for Block Ciphers and Hash Functions*, ISIT 2013.

[2] **P.S.L.M. Barreto and V. Rijmen**: *The Anubis Block Cipher*, In First Open NESSIE Workshop, KU-Leuven, Submission to NESSIE (2000)

[3] **P.S.L.M. Barreto and V. Rijmen**: *The Khazad Legacy-Level Block Cipher*, In First Open NESSIE Workshop, KU-Leuven, Submission to NESSIE (2000)

[4] **P.S.L.M. Barreto and V. Rijmen**:, *The Whirlpool hashing function*, Primitive submitted to NESSIE (September 2000) (Revised May 2003).

[5] **V. D. Belousov**: *Osnovi teorii kvazigrup i lup*, "Nauka", Moskva, 1967.

[6] **T. P. Berger**: *Construction of Recursive MDS Diffusion Layers from Gabidulin Codes*, In INDOCRYPT 2013, LNCS **8250**, (2013), 274 – 285.

[7] **W. C. Brown**: *Matrices over commutative rings*, (1993), Chapman & Hall Pure and Applied Mathematics Series, CRC Press.

[8] **E. Couselo, S. Gonsales, V. Markov, A. Nechaev**: *Recursive MDS- codes and recursively differentiable quasigroup*, Discret. Mat. **10(2)**, (1998), 3 – 29.

[9] **E. Couselo, S. Gonsales, V. Markov, A. Nechaev**: *The parameters of recursive MDS-codes*, Discret. Mat. **12(4)**, (2000), 3 – 24.

[10] **J. Daemen**: *Cipher and Hash Function Design Strategies based on linear and dierential cryptanalysis*, PhD thesis, K.U.Leuven, (1995).

[11] **J. Daemen, V. Rijmen**:*The Design of Rijndael: AES - The Advanced Encryption Standard*, Information Security and Cryptography, Springer-Verlag, (2002).

[12] **J. Daemen, L. Knudsen, V. Rijmen**:*The Block Cipher Square*, In E. Biham, editor, Fast Software Encryption. Proceedings, LNCS **1267**, (1997), 149 – 165.

[13] **S. M. Dehnavi, A. Mahmoodi Rishakani, M. R. Mirzaee Shamsabad, H. Maimani, E. Pasha**: *Construction of New Families of MDS Diffusion Layers*, IACR Cryptology ePrint Archive 2014: 11, (2014).

[14] **S. T. Dougherty, J.-L. Kim, H. Kulosman**: *MDS codes over finite principal ideal rings*, Designs, Codes and Cryptography **50(1)**, (2009), Springer US, 77 – 92.

[15] **X.-D. Dong, C. B. Son, E. Gunawan**: *Matrix characterization of MDS linear codes over modules*, Linear Algebra and its Applications, **277(1-3)**, (1998), Elsevier, 57 – 61.

[16] **D.G. Filho, P.S.L.M. Barreto, V. Rijmen**: *The Maelstrom-0 hash function*, In: SBSeg 2006 (2006)

[17] **H. Feistel**: *Cryptography and computer privacy*, Scientific American, **228** (1973), no. 5, 15 – 23.

[18] **E. M. Gabidulin**: *Theory of codes with maximum rank distance*, Problems of Information Transmission (English translation of Problemy Peredachi Informatsii) **21(1)**, (1985).

[19] **P. Gauravaram, L. R. Knudsen, K. Matusiewicz, F. Mendel, C. Rechberger, M. Schlffer, S.S. Thomsen**: *Gr∅stl a SHA-3 candidate* Submission to NIST, (2008).

[20] **J. Guo, T. Peyrin, A. Poschmann**: *The PHOTON family of lightweight hash functions*, In: Rogaway, P. (ed.) Crypto 2011, LNCS **6841**, (2011), 222 – 239.

[21] **J. Guo, T. Peyrin, A. Poschmann, M. J. B. Robshaw**: *The LED block cipher*, In: Preneel, B. and Takagi, T. (Eds.) CHES 2011, LNCS **6917**, (2011), 326 – 341.

[22] **K. C. Gupta, I. G. Ray**: *On Constructions of MDS Matrices from Companion Matrices for Lightweight Cryptography*, In: A. Cuzzocrea, C. Kittl, D. E. Simos, E. Weippl, L. Xu (Eds.) CD-ARES Workshops 2013. LNCS **8128**, (2013), Springer, Heidelberg, 2943.

[23] **A.S. Hedayat, N.J.A. Sloane, J. Stufken**: *Orthogonal Arrays: Theory and Applications*, Springer, (1999).

[24] **P. Junod, S. Vaudenay**: *Perfect Diffusion Primitives for Block Ciphers. Building Efficient MDS Matrices*, In SAC'04. LNCS **3357**, (2004), 84 – 99.

[25] **P. Junod, S. Vaudenay**, *FOX: A New Family of Block Ciphers*, In SAC 2004, LNCS **3357**, (2005), 114129.

[26] **F. J. MacWilliams, N. J. A. Sloane**: *The Theory of Error-Correcting Codes*, North-Holland Mathematical Library Series, (1977), North Holland Publishing Co.

[27] **V. Markov, A. Nechaev**: *Generalized BCH-theorem and linear recursive MDS-codes*, ACCT (2010).
[28] **V. Rijmen, B. Preneel, A. Bosselaers, E. DeWin**: *The Cipher SHARK*, In D. Gollmann, editor, Fast Software Encryption. Proceedings, LNCS **1039**, (1996), 99 − 111.
[29] **M. Sajadieh, M. Dakhilalian, H. Mala, P. Sepehrdad**: *Recursive Diffusion Layers for Block Ciphers and Hash Functions*, In: Canteaut, A. (ed.) FSE 2012, LNCS **7549**, (2012), 385 − 401.
[30] **C. P. Schnorr, S. Vaudenay**: *Parallel FFT-hashing*, In R. Anderson, editor, FSE 1993, LNCS **809**, (1994), 149 − 156.
[31] **C. P. Schnorr, S. Vaudenay**: *Black Box Cryptanalysis of hash networks based on multipermutations*, In A. De Santis, editor, Advances of Cryptology - EUROCRYPT'94, LNCS **950**, (1995), 47 − 57.
[32] **J. A. Thas**: *M.D.S. Codes and Arcs in Projective Spaces: A Survey*, Le Matematiche **47(Fasc. II)**, (1992), 315 − 328.
[33] **B. Schneier, J. Kelsey, D. Whiting, D. Wagner, C. Hall, N. Ferguson**: *Twofish: A 128-bit block cipher*, In The First AES Candidate Conference, NIST, (1998).
[34] **C. E. Shannon**: *Communication Theory of Secrecy Systems*, Bell System Technical Journal **28(4)**, (1949).
[35] **T. Shirai, K. Shibutani, T. Akishita, S. Moriai, T. Iwata**: *The 128-bit blockcipher CLEFIA (extended abstract)*, In: A. Biryukov (ed.) FSE 2007, LNCS **4593**, (2007) 181195.
[36] **J. Stern, S. Vaudenay**: *CS-Cipher*, In J. Kelsey, B. Schneier, D. Wagner, C. Hall, editors, FSE 1998, LNCS **1372**, (1998), 189 − 204.
[37] **S. Vaudenay**: *On the Need for Multipermutations: Cryptanalysis of MD4 and SAFER*, In B. Preenel, editor, Fast Software Encryption. Proceedings, LNCS **1008**, (1995), 286 − 297.
[38] **S. Wu, M. Wang, W. Wu**: *Recursive Diffusion Layers for (Lightweight) Block Ciphers and Hash Functions*, In L. R. Knudsen, H. Wu (Eds.) SAC 2013, LNCS **7707**, (2013), Springer Berlin Heidelberg,355 − 371.
[39] **A. A. Zain, B. S. Rajan**: *Algebraic characterization of MDS group codes over cyclic groups*, IEEE Transactions on Information Theory **41(6)**, (1995), 2052 − 2056.

University "Goce Delčev", Faculty of Computer Science,, "Krste Misirkov " bb, 2000, Štip, Republic of Macedonia
E-mail address: aleksandra.mileva@ugd.edu.mk