Lightweight MDS Serial-type Matrices with Minimal Fixed XOR Count

Dylan Toh[1] Jacob Teo[1]
Khoongming Khoo[2] Siang Meng Sim[2,3]

1. NUS High School of Math and Science, Singapore
2. DSO National Laboratories, Singapore
3. Nanyang Technological University, Singapore

9 May 2018 @ Africacrypt 2018

Table of Contents

Table of Contents

Diffusion Matrices

The diffusion layer of a cipher (often expressed as a diffusion matrix) plays the role of diffusion:
spread the internal dependencies.

The diffusion matrix $\mathbf{M}$ of order k is applied to a k-tuple input vector $\mathbf{u}$ to create the diffusion.

Its diffusion power can be quantified by the branch number of a matrix.

$$\mathcal{B}(\mathbf{M}) = \min_{\mathbf{u} \neq \mathbf{0}} (wt(\mathbf{u}) + wt(\mathbf{M}\mathbf{u})),$$

where $wt(\cdot)$ is the number of non-zero components in the vector.

Maximal Distance Separable (MDS) Matrices

Definition (MDS)

An MDS matrix $\mathbf{M}$ of order k is a diffusion matrix with optimal branch number $k+1$. I.e. $\mathcal{B}(\mathbf{M}) = k+1$.

Definition (q-MDS)

A q-MDS matrix $\mathbf{M}$ of order k is a diffusion matrix with optimal branch number $k+1$ when raised to its q-th power.
I.e. $\mathcal{B}(\mathbf{M}^q) = k+1$

q-MDS matrix is also called recursive MDS matrix.

Hardware Implementation of Diffusion Matrices

Let $\mathcal{C}(\mathbf{M})$ and $\mathcal{C}(\alpha)$ be the number of XOR gates needed to implement a matrix and its coefficient respectively.

Example

The AES diffusion matrix $\mathbf{M}_A$ (over $\mathrm{GF}(2^8)$).

$$\begin{pmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{pmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} 2a \oplus 3b \oplus c \oplus d \\ a \oplus 2b \oplus 3c \oplus d \\ a \oplus b \oplus 2c \oplus 3d \\ 3a \oplus b \oplus c \oplus 2d \end{bmatrix}$$

The implementation cost $\mathcal{C}(\mathbf{M}_A) = 4 \cdot (\mathcal{C}(2) + \mathcal{C}(3)) + 4 \cdot 3 \cdot 8$.

These two terms are called the variable cost and the fixed cost respectively.

Fixed Cost of MDS matrices

Proposition

All coefficients of an MDS matrix are non-zero.

Thus, the fixed cost of an MDS matrix of order k over $\mathrm{GF}(2^s)$ is

$$k \cdot (k-1) \cdot s$$

One research direction considers the serial-type matrices that are q-MDS to have an area/clock cycle trade-off to achieve lower fixed cost.

Table of Contents

Table of Contents

Linear Feedback Serial (LFS) Matrices

An example of serial-type matrices is the serial matrices (for clarity purposes, we call them LFS matrices) that is used in the family of hash functions PHOTON [GP+11].

Definition

An LFS matrix $\mathbf{L} = LFS(z_0, z_1, ..., z_{k-1})$ of order k is a matrix of the following form:

$$\mathbf{L}_{ij} = \begin{cases} z_j, & i = k-1 \\ 1, & i+1 = j \\ 0, & \text{otherwise.} \end{cases} \qquad e.g. \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ z_0 & z_1 & z_2 & z_3 \end{pmatrix}$$

Area/Clock Cycle Trade-off

Example

Let $\mathbf{L}_4 = LFS(z_0, z_1, z_2, z_3)$ be a diffusion matrix over $\mathrm{GF}(2^8)$.

$$\begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ z_0 & z_1 & z_2 & z_3 \end{pmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} b \\ c \\ d \\ z_0 a \oplus z_1 b \oplus z_2 c \oplus z_3 d \end{bmatrix}$$

The implementation cost $\mathcal{C}(\mathbf{L}_4) = \sum \mathcal{C}(z_i) + 3 \cdot 8$.

Suppose $\mathbf{L}_4$ is 4-MDS, one can implement (costs about $\frac{1}{4}$ of MDS matrices) and reuse the circuit to update the input vector 4 times (4 clock cycles) to achieve the MDS property.

Fixed Cost of LFS matrices

Theorem

If an LFS matrix of order k is k-MDS, then $z_i \neq 0$ for all i.

Thus, the fixed cost of a k-MDS LFS matrix of order k over $\mathrm{GF}(2^s)$ is

$$(k-1)\cdot s$$

Can we achieve even lower fixed cost?

Table of Contents

Diagonal-Serial Invertible (DSI) Matrices

We propose a new serial-type matrix:

Definition

A DSI matrix $\mathbf{D}$ of order k is determined by 2 vectors, $\mathbf{a} = (a_1, a_2, \ldots, a_{k-1}, a_k)$, where a_i's are non-zero, and $\mathbf{b} = (b_1, b_2, \ldots, b_{k-1})$, as follows:

$$D_{ij} = \begin{cases} a_1, & i = 1, j = k \\ a_i, & i = j + 1 \\ b_i, & i = j \leq k - 1 \\ 0, & \text{otherwise.} \end{cases} \qquad e.g. \begin{pmatrix} b_1 & 0 & 0 & a_1 \\ a_2 & b_2 & 0 & 0 \\ 0 & a_3 & b_3 & 0 \\ 0 & 0 & a_4 & 0 \end{pmatrix}$$

Theorem

Every DSI matrix $\mathbf{D} = DSI(\mathbf{a}, \mathbf{b})$ *is invertible.*

Sparse DSI Matrices

Definition

A DSI matrix **D** of order k is **sparse** if **b** has:
$b_{2l} = 0$, where $1 \leq l < \left\lfloor \frac{k}{2} \right\rfloor$.

Example

Sparse DSI matrix $\mathbf{D}_4$ over $\mathrm{GF}(2^8)$:

$$\begin{pmatrix} b_1 & 0 & 0 & a_1 \\ a_2 & 0 & 0 & 0 \\ 0 & a_3 & b_3 & 0 \\ 0 & 0 & a_4 & 0 \end{pmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} b_1 a \oplus a_1 d \\ a_2 a \\ a_3 b \oplus b_3 c \\ a_4 c \end{bmatrix}$$

The implementation cost $\mathcal{C}(\mathbf{D}_4) = \sum \mathcal{C}(a_i) + \sum \mathcal{C}(b_i) + 2 \cdot 8$.

Fixed Cost of Sparse DSI matrices

Corollary

Sparse DSI matrix of order k can potentially be k-MDS.

Thus, the fixed cost of a k-MDS sparse DSI matrix of order k over $\mathrm{GF}(2^s)$ is

$$\left\lceil \frac{k}{2} \right\rceil \cdot s$$

Matrix type	k-MDS sparse DSI		k-MDS LFS		MDS
Fixed cost	$\lceil \frac{k}{2} \rceil \cdot s$	$<$	$(k-1) \cdot s$	$<$	$k \cdot (k-1) \cdot s$

Table of Contents

Table of Contents

Circuit of Sparse DSI and LFS Matrices

Expressing the matrices as the following implementation circuits:

$$\begin{pmatrix} b_1 & 0 & 0 & a_1 \\ a_2 & 0 & 0 & 0 \\ 0 & a_3 & b_3 & 0 \\ 0 & 0 & a_4 & 0 \end{pmatrix} \qquad \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ z_0 & z_1 & z_2 & z_3 \end{pmatrix}$$

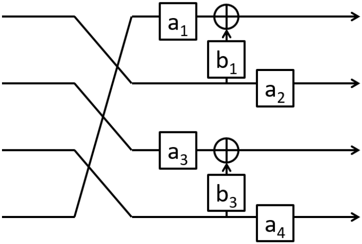

Figure: Sparse DSI

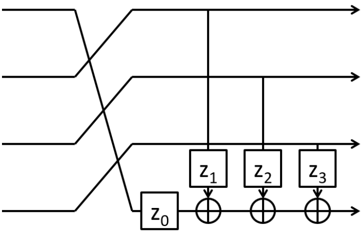

Figure: LFS

Saving Variable Cost for Sparse DSI Matrices

We can save some variable cost if there are identical coefficients:

$$\begin{pmatrix} b_1 & 0 & 0 & a_1 \\ a_2 & 0 & 0 & 0 \\ 0 & a_3 & b_3 & 0 \\ 0 & 0 & a_4 & 0 \end{pmatrix}$$

Suppose we have $a_2 = b_1$, then we can save $\mathcal{C}(b_1)$.

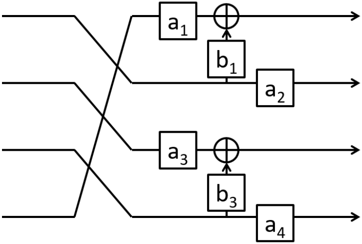

Figure: Sparse DSI

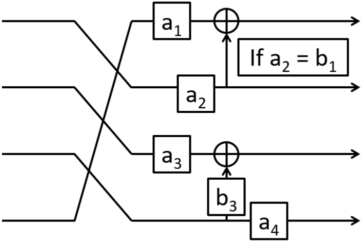

Figure: Sparse DSI saving cost

Saving Variable Cost for LFS Matrices

Similarly for LFS matrices:

$$\begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ z_0 & z_1 & z_2 & z_3 \end{pmatrix}$$

Suppose we have $z_2 = z_1$, then we can save $\mathcal{C}(z_2)$.

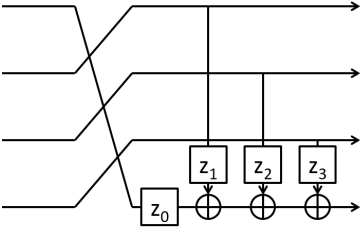

Figure: LFS

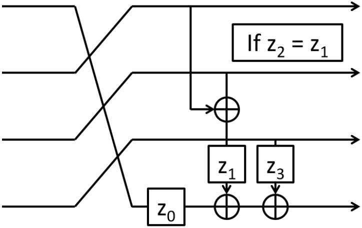

Figure: LFS saving cost

Total XOR count of various matrices

matrices of order k over $GF(2^s)$	XOR count of the entire matrix
Sparse DSI	$\sum \mathcal{C}(a_i) + \sum_{b_i \neq a_{i+1}} \mathcal{C}(b_i) + \lceil k/2 \rceil \cdot s$
LFS	$\sum_{z_i \mid \forall j<i, z_i \neq z_j} \mathcal{C}(z_i) + (k-1) \cdot s$
MDS	$\sum \mathcal{C}(m_i) + k \cdot (k-1) \cdot s$

where m_i's are the entries (k^2 elements) of an MDS matrix.

We also show that the inverse of DSI and LFS matrices (backward) has the same implementation cost as the forward direction.

Table of Contents

Comparison of Diffusion Matrices

We found new lightweight sparse DSI and LFS matrices of order 4 to 8 over $\mathrm{GF}(2^4)$ and $\mathrm{GF}(2^8)$.

Implementation cost of diffusion matrices over $\mathrm{GF}(2^8)$

k	Matrix Type	Forward	Backward	Ref.
4	Circulant	$48 + 4 \cdot 24 = \mathbf{144}$	$212 + 4 \cdot 24 = \mathbf{308}$	AES
4	LFS	$9 + 24 = \mathbf{33}$		[KP+14]
4	8-MDS LFS	$3 + 24 = \mathbf{27}$		[SS+17]
4	Sparse DSI	$6 + 16 = \mathbf{22}$		This Paper
5	IMDS left-circ.	$165 + 5 \cdot 32 = \mathbf{325}$		[LS16]
5	Left-circulant	$50 + 5 \cdot 32 = \mathbf{210}$	$290 + 5 \cdot 32 = \mathbf{450}$	[LS16]
5	Sparse DSI	$7 + 24 = \mathbf{31}$		This Paper
6	LFS A_{288}	$17 + 40 = \mathbf{57}$		PHOTON
6	Sparse DSI	$7 + 24 = \mathbf{31}$		This Paper

Table of Contents

Trade-off for Diffusion Matrices

In comparison with MDS diffusion matrices of order k, k-MDS LFS matrices have a fair trade-off between the area and clock cycle.

$$\text{MDS matrix} \xrightarrow[k\cdot(k-1)\downarrow(k-1)\ s\text{-bit XORs}]{1\uparrow k \text{ clock cycles}} k\text{-MDS LFS matrix}$$

Our k-MDS sparse DSI (sDSI) matrices achieve a trade surplus.

$$\text{MDS matrix} \xrightarrow[k\cdot(k-1)\downarrow\lceil\frac{k}{2}\rceil\ s\text{-bit XORs}]{1\uparrow k \text{ clock cycles}} k\text{-MDS sDSI matrix}$$

A natural question to ask:
Can we achieve even lower fixed cost?
We look at diffusion matrices of order 4.

OXS Matrix of Order 4

Sparse DSI matrix has 2 s-bit XORs, thus we consider matrix with only 1 s-bit XOR, so-called One XOR Serial (OXS) matrix.

$$\begin{array}{c} \text{4-MDS} \\ \text{sDSI matrix} \end{array} \xrightarrow[2 \downarrow 1\ s\text{-bit XORs}]{4 \uparrow \leq 8 \text{ clock cycles}} \begin{array}{c} q\text{-MDS} \\ \text{OXS matrix} \end{array}$$

We consider $q \leq 8$, otherwise it is a bad trade-off.

Theorem

There does not exist OXS matrix of order 4 that is q-MDS, where $q \leq 8$.

Therefore, we conclude that our sparse DSI matrix of order 4 has the minimal fixed cost.

Conclusion

- Introduce a new serial-type matrix called Diagonal-Serial Invertible (DSI) matrix with sparse property
- k-MDS sparse DSI matrix has a positive trade-off between the hardware area and clock cycle
- Found new lightweight serial-type matrices
- Prove that our sparse DSI matrix of order 4 has the minimal fixed cost

Reference

[GP+11]—J. Guo, T. Peyrin and A. Poschmann. The PHOTON Family of Lightweight Hash Functions. In CRYPTO 2011.

[KP+14]—K. Khoo, T. Peyrin, A. Poschmann and H. Yap. FOAM: Searching for Hardware-Optimal SPN Structures and Components with a Fair Comparison. In CHES 2014.

[LS16]—M. Liu and S.M. Sim. Lightweight MDS generalized circulant matrices. In FSE 2016.

[SS+17]—S. Sarkar, H. Syed, R. Sadhukhan and D. Mukhopadhyay. Lightweight design choices for LED-like block ciphers. In INDOCRYPT 2017.

Thank you. :)