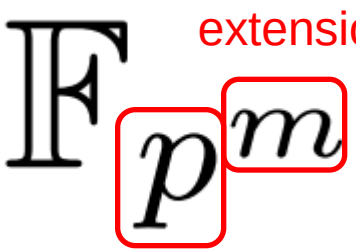
Mixed Bases for Efficient Inversion in $F((2^2)^2)^2$ and Conversion Matrices of SubBytes of AES

Information Transfer Laboratory
Okayama University, Japan
Yasuyuki Nogami, Kenta Nekado, Tetsumi Toyota,
Naoto Hongo, and Yoshitaka Morikawa

Research background (1)

- **Odd characteristic extension field**
  - Cryptographic applications
    - Pairing-based cryptography with elliptic curves
    - Efficient arithmetic operations
    - Using several kinds of bases in mixture

      they respectively have efficient calculations
  - Modular polynomials
    - Irreducible binomials, irreducible trinomials
    - Cyclotomic polynomials
  - Bases
    - **P**olynomial basis : *multiplication*
    - **N**ormal basis : *Frobenius mapping*
      - Gauss period normal basis (GNB)



Research background (2)

- **Itoh-Tsujii inversion algorithm (ITA)**
  - Multiplications by conjugates
  - Frobenius mappings

$$X^{-1} = \left(X^{p} \cdots X^{p^{m-1}}\right)\left(X \cdot X^{p} \cdots X^{p^{m-1}}\right)^{-1}$$

subfield inversion

  - In the case of characteristic two
    - Frobenius mapping is equivalent to squaring.

$$X^{-1} = \left(X^{2} \cdots X^{2^{m-1}}\right)$$

Normal bases efficiently work for ITA.

Research background (3)

- **In the case of pairing-based cryptographies…**
  - Binary extension fields $\mathbb{F}_{2^m}$
  - Ternary extension fields $\mathbb{F}_{3^m}$
  - Other odd characteristic extension fields
    - Parameters
      - p : 160 bits - 256 bits
      - m : 2,3,4, …, 20
    - Research targets
      - Vector multiplication algorithm
      - Exponentiation and scalar multiplication for rational points

$$\mathbb{F}_{p^m}$$

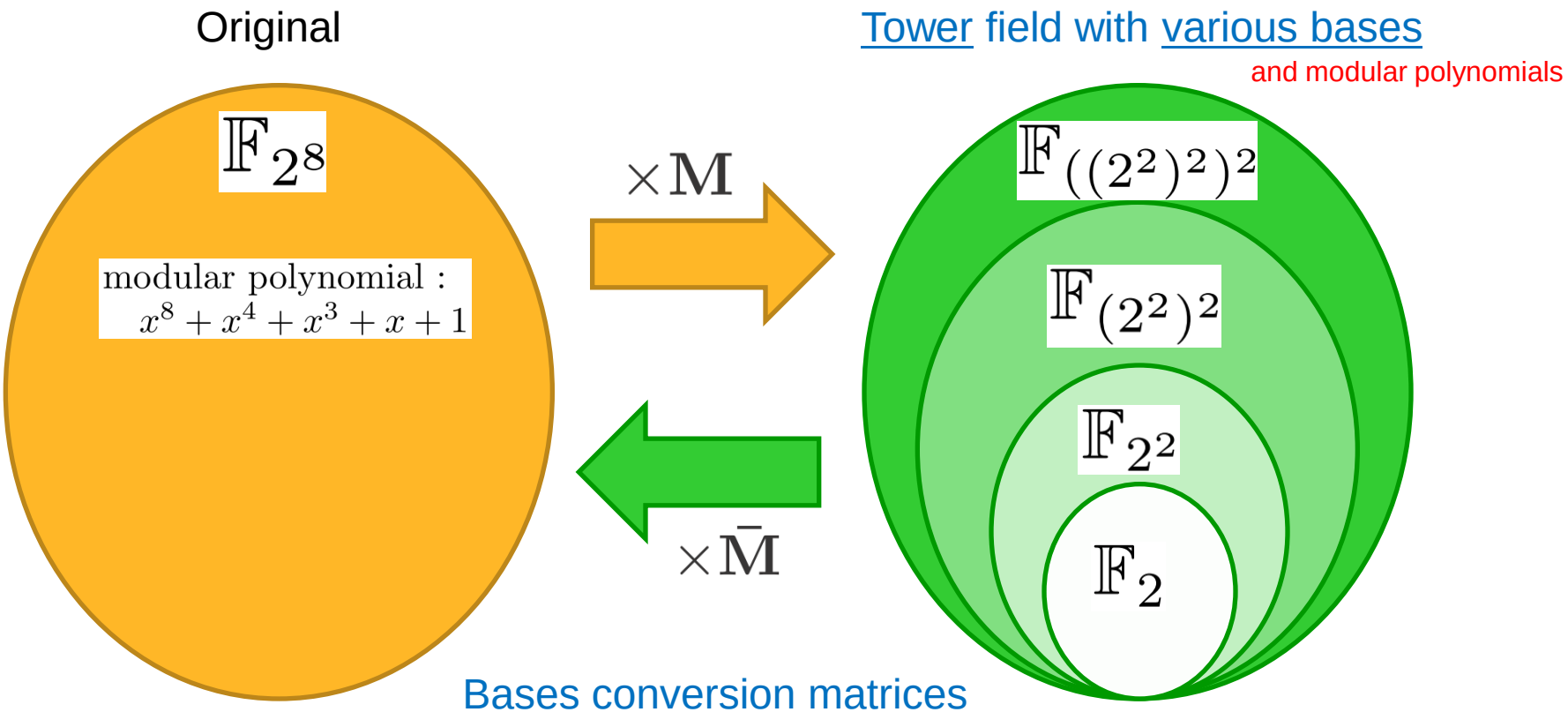
motivation

- For using several kinds of bases in mixture
  - Bases conversion matrices are required.

our previous work : GNB-based derivation

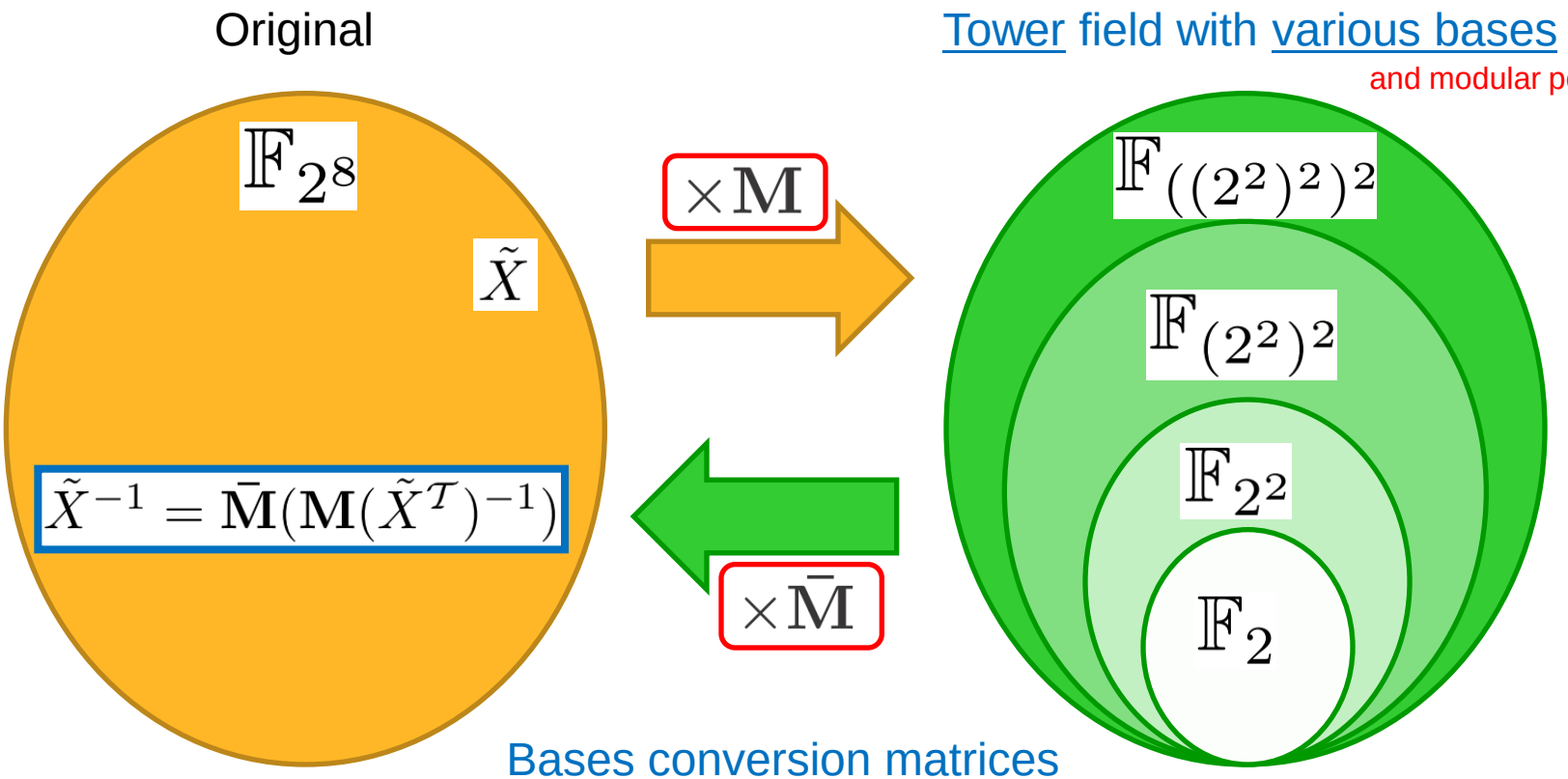
Then, as an application …

- $\mathbb{F}_{2^8}$ **inversion for SubBytes of AES**



Then, as an application …

- $\mathbb{F}_{2^8}$ **inversion for SubBytes of AES**



However, …

GNB-based derivation :

``Finding a Basis Conversion Matrix via Prime Gauss Period Normal Basis,"
Y. Nogami, R. Namba, and Y. Morikawa, IEICE Trans., E92-A, no. 6, pp. 1500 - 1507, 2009.

- **Condition for the existence of GNBs**

$$4p \mid m(p-1)$$

– When $p = 2$ and $m = 8$ for AES, it is not satisfied…

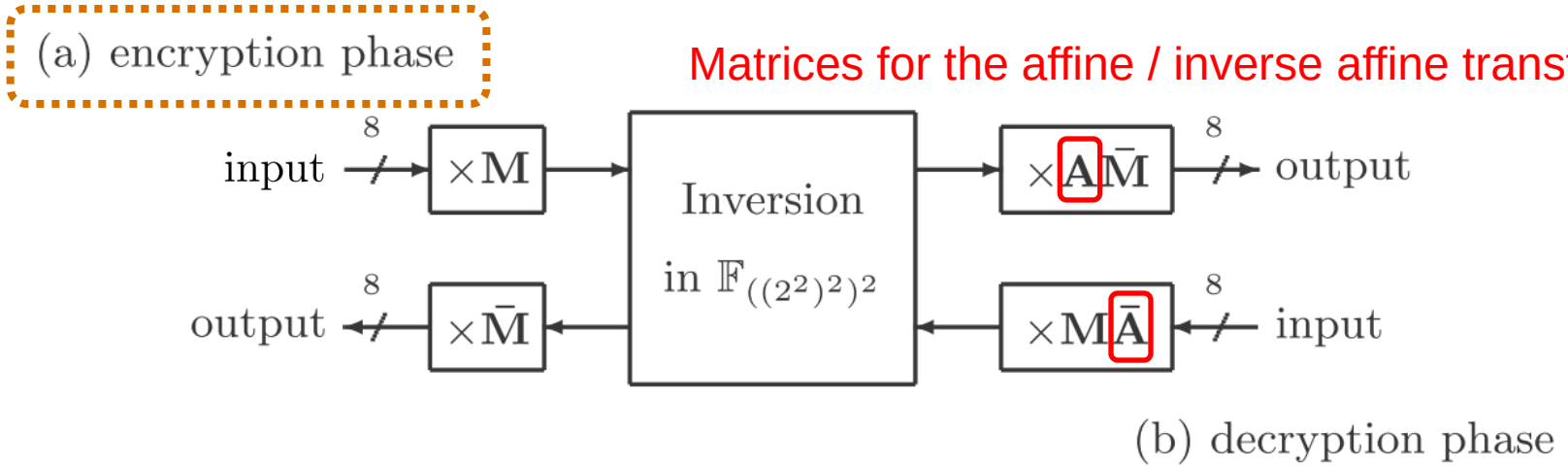
No problem. Full search !

$$2^8 = 256$$ the order is very small !

– Once I had lost the motivation, with respect to the conjugates, 8 variants existed for the pair of conversion matrices $\mathbf{M}$, $\bar{\mathbf{M}}$

The detail of AES

- **AES also needs affine / inverse affine transforms**



8 variants of $\mathbf{M}$, $\bar{\mathbf{M}}$

In the encryption phase,

there are 8 variants of $\mathbf{M}$ and $\mathbf{A}\bar{\mathbf{M}}$.

Previous works

- **According to some previous works, there are**

432 towering constructions of $\mathbb{F}_{((2^2)^2)^2}$.

  - For each, there are 8 variants of $\mathbf{M}$, $\mathbf{A}\bar{\mathbf{M}}$.

For example,

$$\mathbf{M} = \begin{bmatrix} 1&0&1&0&0&1&0&0 \\ 1&0&0&0&1&0&1&0 \\ 0&0&1&1&1&1&0&0 \\ 0&1&0&1&0&1&0&0 \\ 1&1&0&1&0&1&0&0 \\ 1&0&1&1&0&0&0&1 \\ 0&1&0&0&0&1&1&1 \\ 0&0&1&0&1&0&1&0 \end{bmatrix}, \quad \mathbf{A}\bar{\mathbf{M}} = \begin{bmatrix} 1&1&0&1&1&1&0&0 \\ 0&1&1&1&0&1&1&1 \\ 0&1&0&0&0&1&1&0 \\ 1&0&0&1&1&1&1&1 \\ 0&1&1&0&0&0&1&0 \\ 1&0&0&0&0&0&1&0 \\ 0&0&0&1&1&0&1&0 \\ 0&1&1&1&0&0&1&1 \end{bmatrix}.$$

efficient / inefficient !?

Conversion matrices

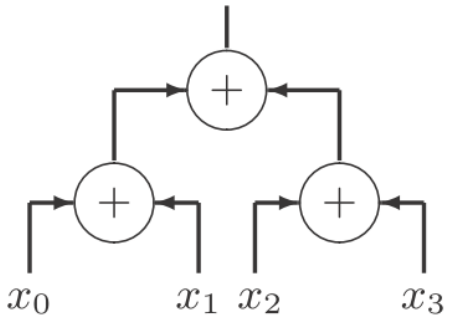
- **Binary vectors are multiplied by the matrices.**

Example : row vectors

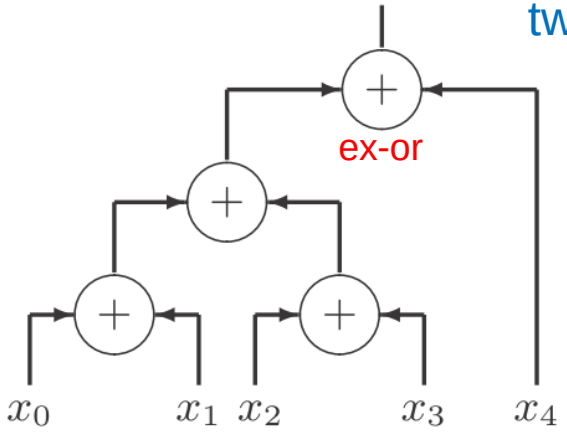
$$(1,1,1,1,0,0,0,0)(x_0,x_1,x_2,x_3,x_4,x_5,x_6,x_7)^{\mathcal{T}}, \quad (14a)$$

$$(1,1,1,1,1,0,0,0)(x_0,x_1,x_2,x_3,x_4,x_5,x_6,x_7)^{\mathcal{T}}. \quad (14b)$$

two inputs logic



Eq. (14a)



Eq. (14b)

The Hamming weights of row vectors are preferred to be smaller than 5.

Hamming weights of row vectors

- **Efficient conversion matrices**

good or bad !?

The **maximum** Hamming weight of row vectors is …

$$\mathbf{M} = \begin{bmatrix} 1&0&1&0&0&1&0&0 \\ 1&0&0&0&1&0&1&0 \\ 0&0&1&1&1&1&0&0 \\ 0&1&0&1&0&1&0&0 \\ 1&1&0&0&1&0&0&0 \\ 1&0&1&1&0&0&0&1 \\ 0&1&0&0&0&1&1&1 \\ 0&0&1&0&1&0&1&0 \end{bmatrix}.$$

good

smaller than 5

$$\mathbf{A}\bar{\mathbf{M}} = \begin{bmatrix} 1&1&0&1&1&1&0&0 \\ 0&1&1&1&0&1&1&1 \\ 0&1&0&0&0&1&1&0 \\ 1&0&0&1&1&1&1&1 \\ 0&1&1&1&0&0&1&0 \\ 1&0&0&0&0&0&1&0 \\ 0&0&0&1&1&0&1&0 \\ 0&1&1&1&0&0&1&1 \end{bmatrix}.$$

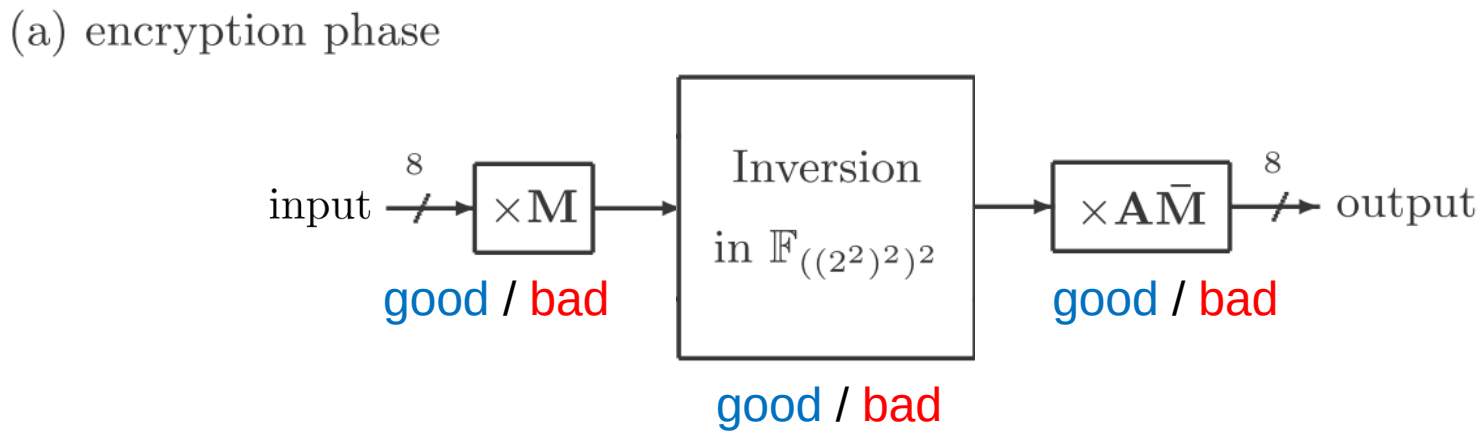
bad

larger than or equal to 5

The 432 constructions **rarely** had good – good conversion matrices.

Notations for the efficiency

- **Notations with good and bad**



Unfortunately,

the rare constructions of good – ? – good conversion matrices

were all good – bad – good constructions …

The rare constructions did not have efficient inversions in $F_{((2^2)^2)^2}$.

good – good – bad construction

- **Sec. 2.4 construction**

  In general, I/O bases are the same.

	basis type
$\mathbb{F}_{2^2}$	**N**ormal
$\mathbb{F}_{(2^2)^2}$	**P**olynomial
$\mathbb{F}_{((2^2)^2)^2}$	**N**ormal

(a) encryption phase

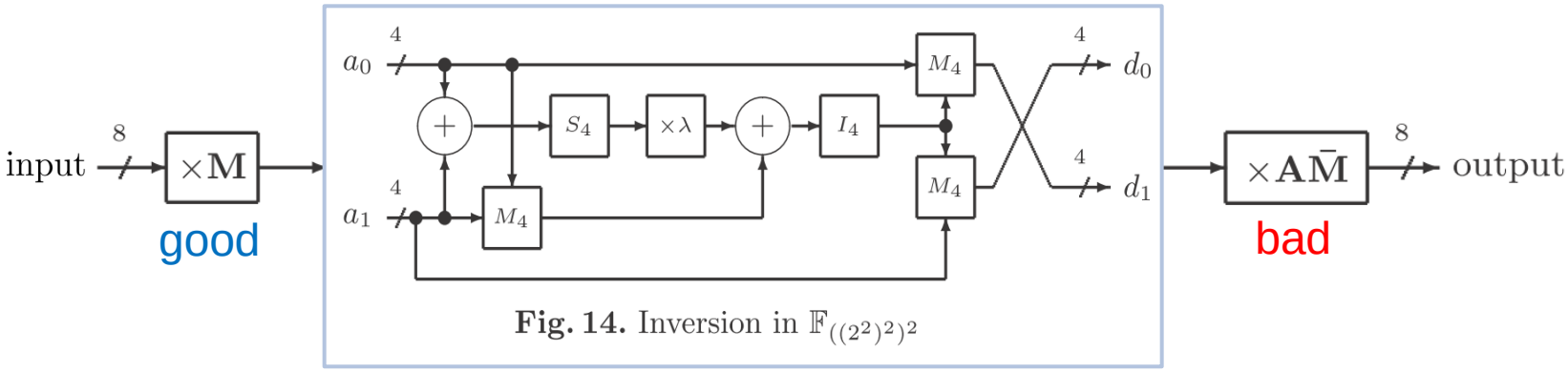


Fig. 14. Inversion in $\mathbb{F}_{((2^2)^2)^2}$

good

Based on this good – good – bad construction,

the idea of *mixed bases* is applied.

Main idea – the first mix –

- **If the output uses other type of basis …**

without loss of efficiency

	basis type
$\mathbb{F}_{2^2}$	**N**ormal
$\mathbb{F}_{(2^2)^2}$	**P**olynomial
$\mathbb{F}_{((2^2)^2)^2}$	**N**ormal

S_4 squaring in $\mathbb{F}_{(2^2)^2}$

M_4 multiplication

I_4 inversion $\times\lambda$ times const.

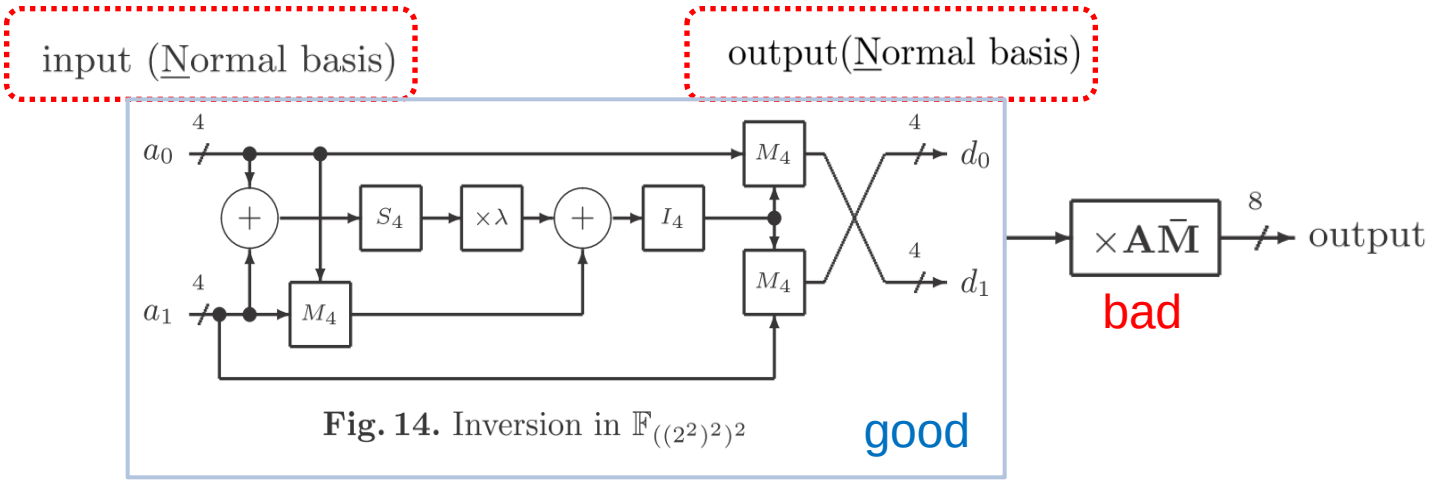


Fig. 14. Inversion in $\mathbb{F}_{((2^2)^2)^2}$

Main idea – the first mix –

- **If the output uses other type of basis**

without loss of efficiency

	basis type
$\mathbb{F}_{2^2}$	**N**ormal
$\mathbb{F}_{(2^2)^2}$	**P**olynomial
$\mathbb{F}_{((2^2)^2)^2}$	**N**ormal / **P**olynomial

the first mix !

the first mix !

Normal input $\{\gamma, \gamma^{16}\}$

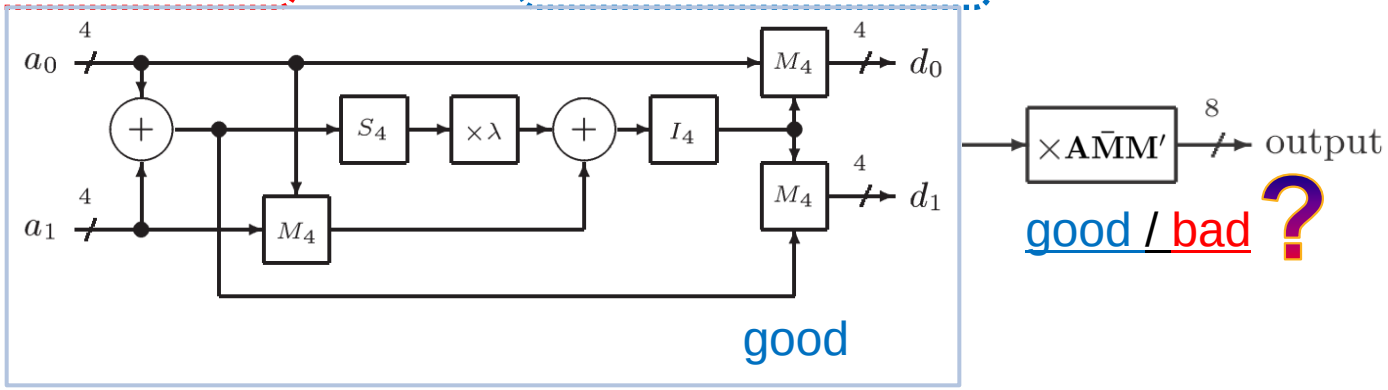
$$A = a_0\gamma + a_1\gamma^{16}, \quad a_0, a_1 \in \mathbb{F}_{(2^2)^2}$$

$$\begin{aligned} A^{-1} &= (AA^{16})^{-1}A^{16} \\ &= \{a_0a_1(\gamma+\gamma^{16})^2 + (a_0^2+a_1^2)\gamma\gamma^{16}\}^{-1}(a_1\gamma + a_0\gamma^{16}) \\ &= \{a_0a_1 + (a_0+a_1)^2\lambda\}^{-1}\{a_0 + (a_0+a_1)\gamma\} \\ &= d_0 + d_1\gamma = D. \end{aligned}$$

Polynomial output $\{1, \gamma\}$

input (Normal basis)

output (Polynomial basis)



good / bad ?

good

Main idea – the first mix –

- **If the output uses other type of basis …**

without loss of efficiency

	basis type
$\mathbb{F}_{2^2}$	**N**ormal
$\mathbb{F}_{(2^2)^2}$	**P**olynomial
$\mathbb{F}_{((2^2)^2)^2}$	**N**ormal / **P**olynomial

the first mix !

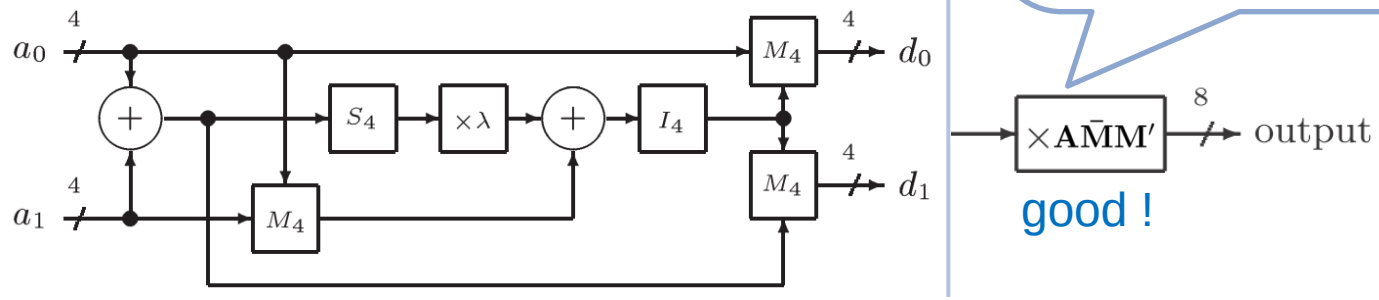
The maximum Hamming weight of row vectors is smaller than 5.

$$\mathbf{A}\bar{\mathbf{M}}\mathbf{M}' = \begin{bmatrix} 0&0&0&1&1&1&0&1 \\ 0&0&0&0&0&1&1&1 \\ 0&0&1&0&0&1&0&0 \\ 0&1&1&0&1&0&0&1 \\ 0&1&0&0&0&1&1&0 \\ 1&0&1&0&1&0&0&0 \\ 1&0&1&1&0&0&0&1 \\ 0&1&0&0&0&1&1&1 \end{bmatrix}, \quad (20a)$$

good

input (Normal basis)

output (Polynomial basis)



good !

Another good conversion matrix could follow.

Extension of the idea – the second mix –

- **If normal bases can be partially applied …**

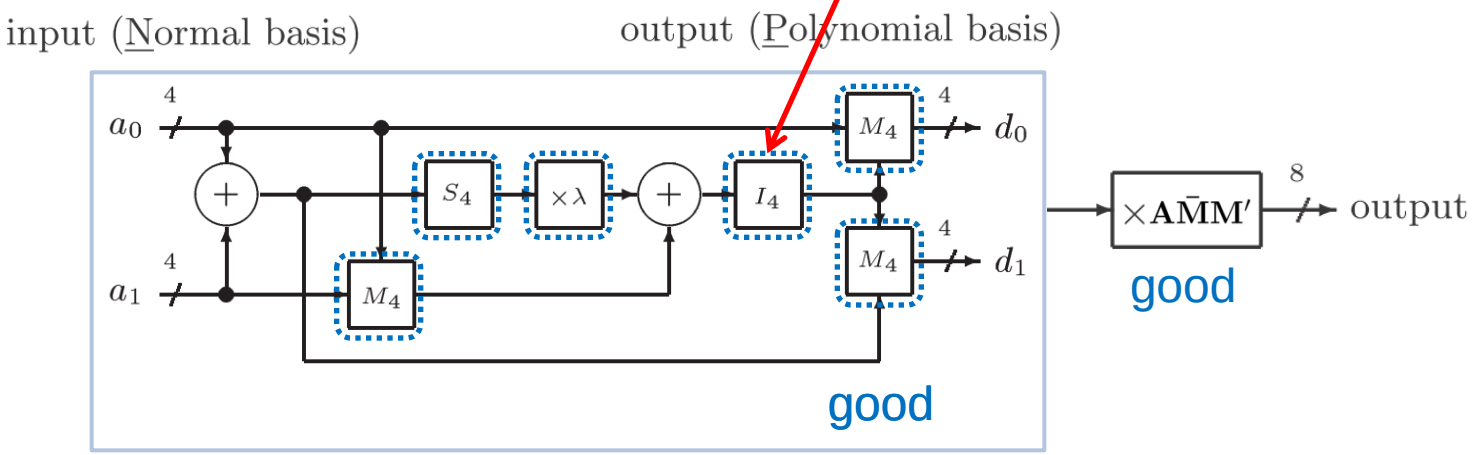
	basis type
$\mathbb{F}_{2^2}$	**N**ormal
$\mathbb{F}_{(2^2)^2}$	**P**olynomial
$\mathbb{F}_{((2^2)^2)^2}$	**N**ormal / **P**olynomial

S_4 squaring in $\mathbb{F}_{(2^2)^2}$

M_4 multiplication

I_4 inversion $\times\lambda$ times const.

Normal basis is efficient for inversion…



Extension of the idea – the second mix –

- **If normal bases can be partially applied …**

	basis type
$\mathbb{F}_{2^2}$	**N**ormal
$\mathbb{F}_{(2^2)^2}$	**P**olynomial
$\mathbb{F}_{((2^2)^2)^2}$	**N**ormal / **P**olynomial

Normal basis is partially used *in mixture*.

the second mix !

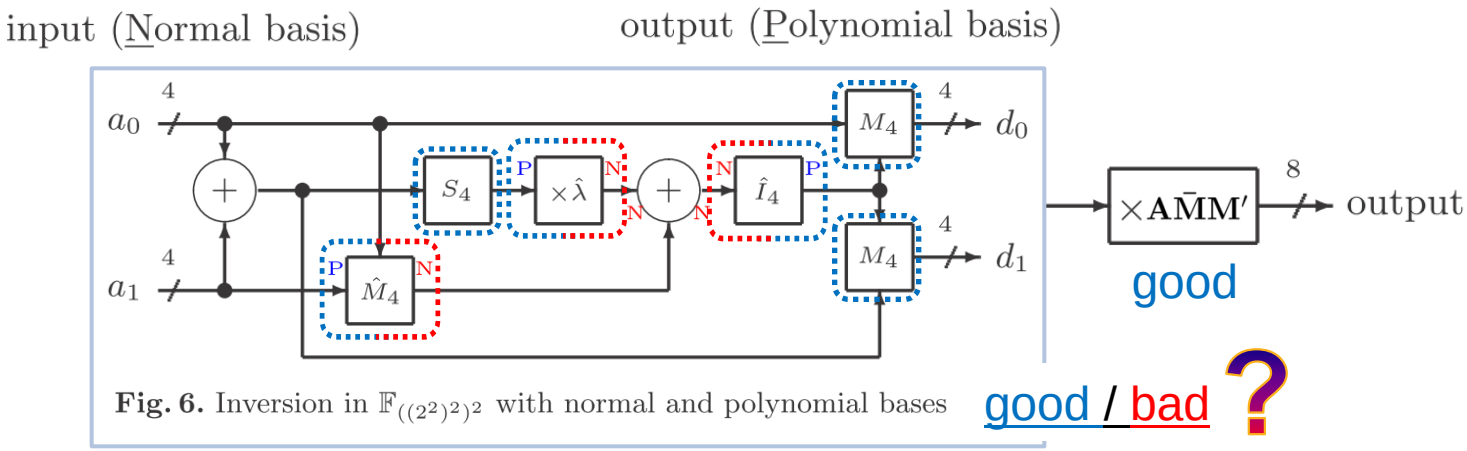


Fig. 6. Inversion in $\mathbb{F}_{((2^2)^2)^2}$ with normal and polynomial bases

good / bad ?

Extension of the idea – the second mix –

- **If normal bases can be partially applied …**

	basis type
$\mathbb{F}_{2^2}$	**N**ormal
$\mathbb{F}_{(2^2)^2}$	**P**olynomial
$\mathbb{F}_{((2^2)^2)^2}$	**N**ormal / **P**olynomial

Normal b

the second mix !

Polynomial input – **P**olynomial output

Normal input $\{\beta, \beta^4\}$

$$A = a_0\beta + a_1\beta^4$$

$$\begin{aligned} A^{-1} &= (AA^4)^{-1}A^4 \\ &= \left\{(a_0\beta + a_1\beta^4)(a_1\beta + a_0\beta^4)\right\}^{-1}(a_1\beta + a_0\beta^4) \\ &= \left\{a_0a_1 + (a_0+a_1)^2\alpha\right\}^{-1}((a_0+a_1) + a_0\beta) \\ &= d_0 + d_1\beta = D. \end{aligned}$$

Polynomial output $\{1, \beta\}$

input (Normal basis) output (Pc

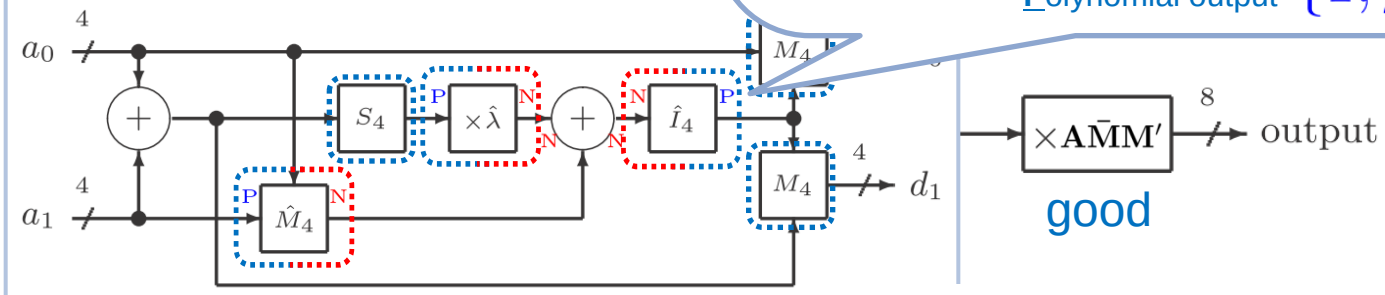


Fig. 6. Inversion in $\mathbb{F}_{((2^2)^2)^2}$ with normal and polynomial bases

better !

Conclusion and future works

- **Mixed bases technique**

  good – good – bad construction

  good – better – good construction

- **I would like to thank to the anonymous reviewers.**
  - Security issue for such special conversion matrices
  - Total approach together with the decryption phase

- **Other towering fields**
  - For example, $\mathbb{F}_{(2^4)^2}$

Thank you for your attention !