Computational work for some TU-based permutations

Dmitry Trifonov[1] and Denis Fomin[2]

[1] Academy of Cryptography of Russian Federation
[2] HSE University

Low-resource nonlinear bijective transformations

The most commonly used approaches for evaluating the low resourceless of nonlinear transformations are:

- nonlinear complexity
- gate-complexity
- bitslice-complexity
- combinational complexity

There are three main approaches to construct permutations with given performance characteristics:

- full search (with optimizers)
- heuristic search
- use of simple algebraic (monomial) permutations

Metrics

Definition

The combinational complexity of a function f in the basis Ω, denoted by $C_{\Omega}(f)$, is the minimal number of elements of the basis Ω sufficient for realization of the function f by a logic circuit.

Definition

The circuit depth (or just depth) of the function f in the basis Ω, denoted by $D_{\Omega(f)}$, s the number of logical elements located on the longest oriented path of the graph representing the logical scheme.

Remark

In this work we use the following basis: $\Omega = \{\wedge, \vee, \oplus, \neg\}$.

Some bases I

Definition

Let α be an element of the field $\mathbb{F}_{2^n}$ such that the set $\{\alpha^i\}_{i=0}^{m-1}$ is the basis of $\mathbb{F}_{2^n}$ over $\mathbb{F}_{2^k}$. In this case it is said that $\{\alpha^i\}_{i=0}^{m-1}$ is the polynomial basis of the field $\mathbb{F}_{2^n}$ over $\mathbb{F}_{2^k}$ and α is called the generator of the polynomial basis.

Hereinafter a polynomial basis is denoted by Poly.

Definition

Let α be an element of the field $\mathbb{F}_{2^n}$ such that the set $\left\{\alpha^{2^i}\right\}_{i=0}^{m-1}$ is the basis of $\mathbb{F}_{2^n}$ over $\mathbb{F}_{2^k}$. In this case it is said that $\left\{\alpha^{2^i}\right\}_{i=0}^{m-1}$ is the normal basis of the field $\mathbb{F}_{2^n}$ over $\mathbb{F}_{2^k}$ and α is called the generator of the normal basis.

Hereinafter a normal basis is denoted by Norm.

Some bases II

One of the most effective way to reduce combinatorial complexity and depth of functions implementing field operations is to use the field tower theorem as in [1].

The elements of the field $\mathbb{F}_{2^8}$ can be represented as a vector $\mathbb{F}_{2^4}^2$,and the elements of the field $\mathbb{F}_{2^4}$ consider as a vector $\mathbb{F}_{2^2}^2$, etc.

Thus, the elements of the field $\mathbb{F}_{2^8}$ can be represented by vectors from the set $\left(\left(\mathbb{F}_2^2\right)^2\right)^2$.

[1] Yasuyuki Nogami, Kenta Nekado, Tetsumi Toyota, Naoto Hongo, and Yoshitaka Morikawa. Mixed bases for efficient inversion in $GF\left(\left(2^2\right)^2\right)^2$ and conversion matrices of subbytes of aes. IEICE Trans. Fundam. Electron. Commun. Comput. Sci., 94-A(6):1318–1327, 2011.

Important remark

In this work we will evaluate the combinatorial complexity and depth of functions realizing some functions over the field.

Since the type of function directly depends on the basis and the field over which the transformation is considered, let us introduce additional notations:

$$C_{\Omega}\left(f; \mathbb{FF}; \mathrm{Basis}\right)$$

and

$$D_{\Omega}\left(f; \mathbb{FF}; \mathrm{Basis}\right)$$

will denote the combinational complexity and depth of the function f, defined over the field $\mathbb{FF}$ in the basis Basis.

Some classes of permutations I

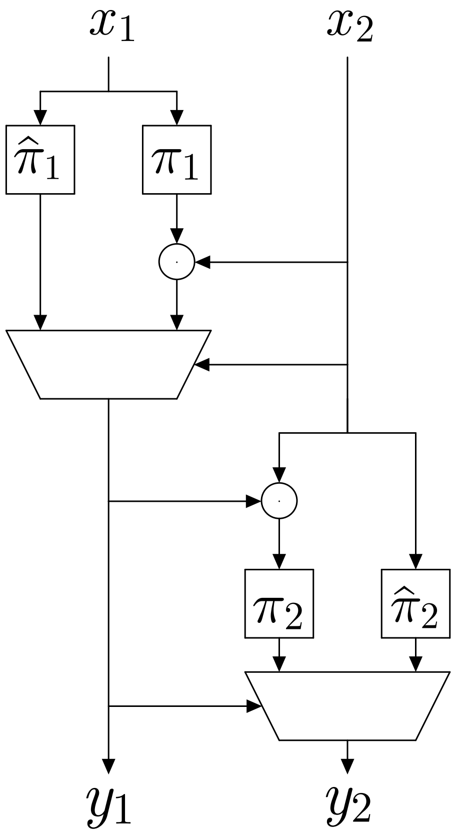


Figure: Type “A” permutation

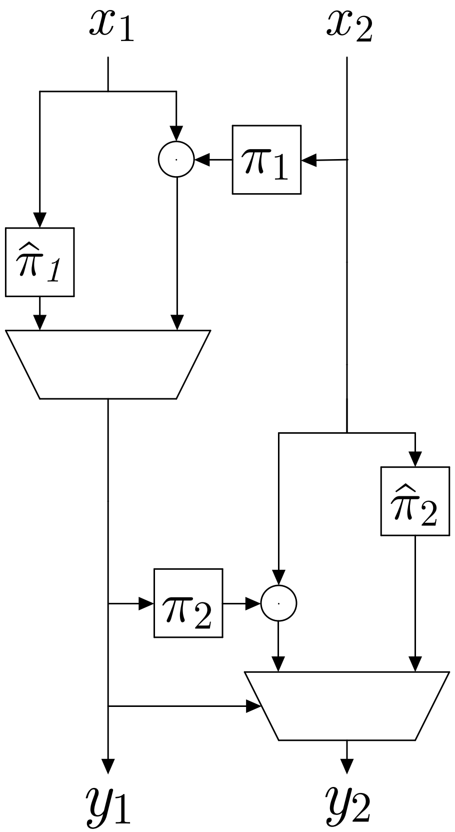


Figure: Type “B” permutation

Some classes of permutations II

Consider a family of permutations, the parameters of which are four degrees: $(\alpha, \beta, \gamma, \delta)$ and permutations $\widehat{\pi}_i$, $i \in \overline{1,2}$: Consider a family of permutations, the parameters of which are four degrees: $(\alpha, \beta, \gamma, \delta)$ and permutations $\widehat{\pi}_i$, $i \in \overline{1,2}$:

$$\begin{gathered} G_1\left(x_1, x_2\right) = y_1 = \begin{cases} x_1^{\alpha} \cdot x_2^{\beta}, & x_2 \neq 0 \\ \widehat{\pi}_1\left(x_1\right), & x_2 = 0 \end{cases}, \\ G_2\left(x_1, x_2\right) = y_2 = \begin{cases} x_1^{\gamma} \cdot x_2^{\delta}, & x_1 \neq 0 \\ \widehat{\pi}_2\left(x_2\right), & x_1 = 0 \end{cases}. \end{gathered} \tag{1}$$

Some classes of permutations III

Such a representation of type “A” and type “B” permutations potentially reduces the depth of the function implementing the permutation.

Moreover, among all known permutations from the proposed families, the following permutation $G(x_1, x_2) = (y_1, y_2)$ from the parametric family «G» has the minimum number of nonlinear transformations used and is set as follows:

1. $x' = x_1^{-1}$
2. $y' = x_2^{-1}$
3. $x'' = x_1 \cdot y'$
4. $y'' = x' \cdot y'$
5. $\texttt{if } x = 0 \texttt{ then } y_1 = y' \texttt{ else } y_1 = y''$
6. $\texttt{if } x = y \texttt{ then } y_2 = x' \texttt{ else } y_2 = x''$

Some classes of permutations IV

Algorithm/Constructing way	N_S	δ_S	$\deg(S)$	AI_{gr}
«AES»	112	4	7	2
«Kuznyechik»	100	8	7	3
<BelT»	102	8	6	3
«SNOW 3G» (SQ)	96	8	5	3
Spectral-linear and specrral-differential methods	104	6	7	3
Permutations from considering classes	**108**	**6**	**7**	**3**
Permutations from[2]	108	6	7	3
Permutations from[3]	108	4	6	3

[2] Cruz Jiménez, R. A. de la. Generation of 8-bit S-Boxes having almost optimal cryptographic properties using smaller 4-bit S-Boxes and finite field multiplication

[3] D. Burov, S. Kostarev and A. Menyachikhin Class of piecewise-monomial mappings: differentially 4-uniform permutations on F28 with graph algebraic immunity 3 exist. CTCrypt'23, 2023

One more important remark

Permutations $\widehat{\pi}_1, \widehat{\pi}_2$ in the parametric families of types “A”, “B” and “G” in [4] is proposed to choose using a heuristic search.

Remark

In this paper an experimental study of affine equivalence classes of $\widehat{\pi}_1, \widehat{\pi}_2$ for the considered parametric families was carried out in the case of construction of the permutation of space $\mathbb{F}_2^8$. This can be done since there is a complete classification of permutations for the space $\mathbb{F}_2^4$. Experimental results have shown that in the vast majority of cases the mentioned permutations belong to only two families of $\mathbb{F}_2^4$ permutations with representatives x^{14} and $x^7 + x^4 + x$. Thus, it is of interest to find the complexity of the mentioned permutations.

[4] Kovrizhnykh M. A. and Fomin D. B. Heuristic algorithm for obtaining permutations with given cryptographic properties using a generalized construction Applied Discrete Mathematics., 57:P. 5–21, 2022.

Comparisons

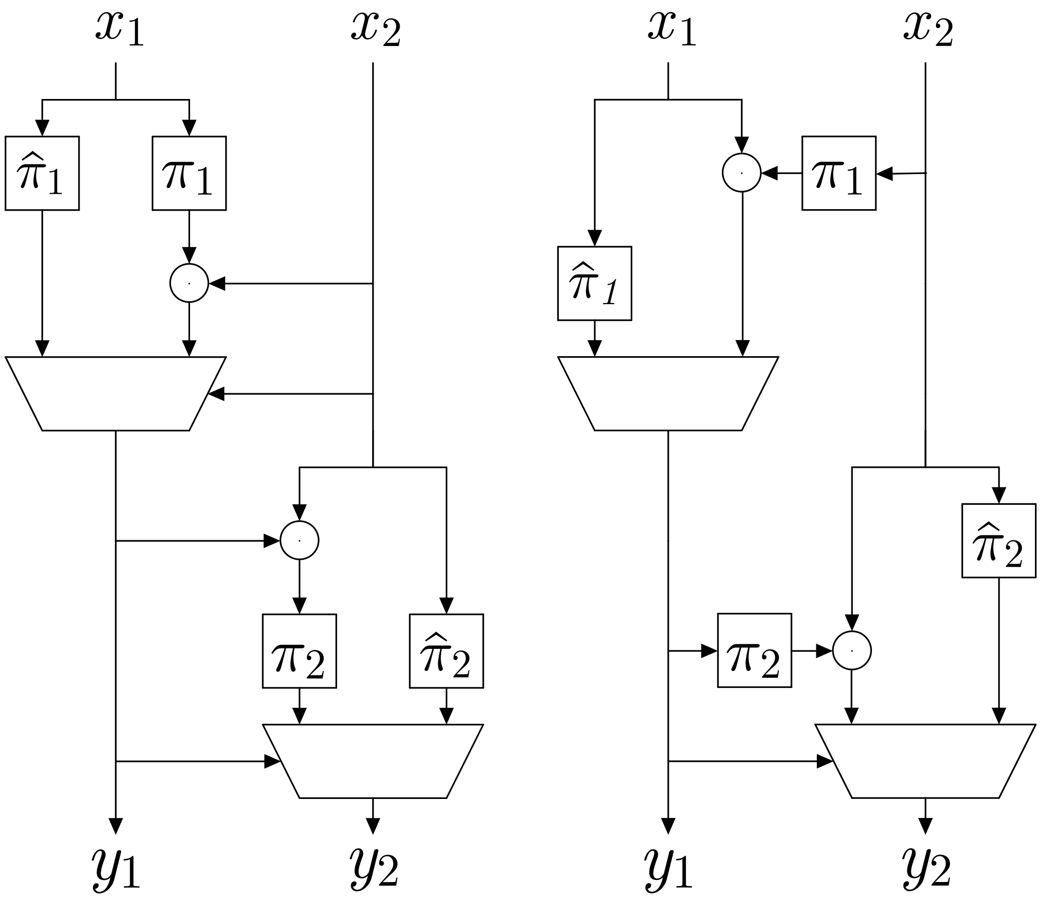


Figure: Type “A” and “B” permutations

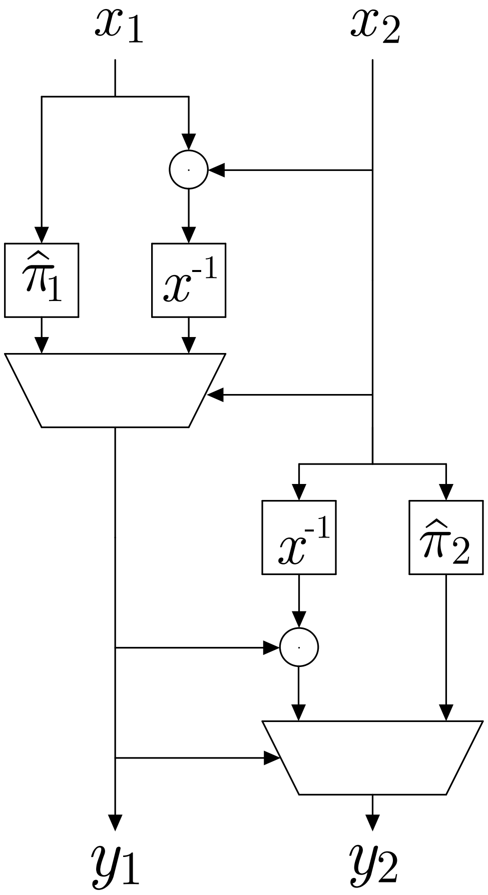
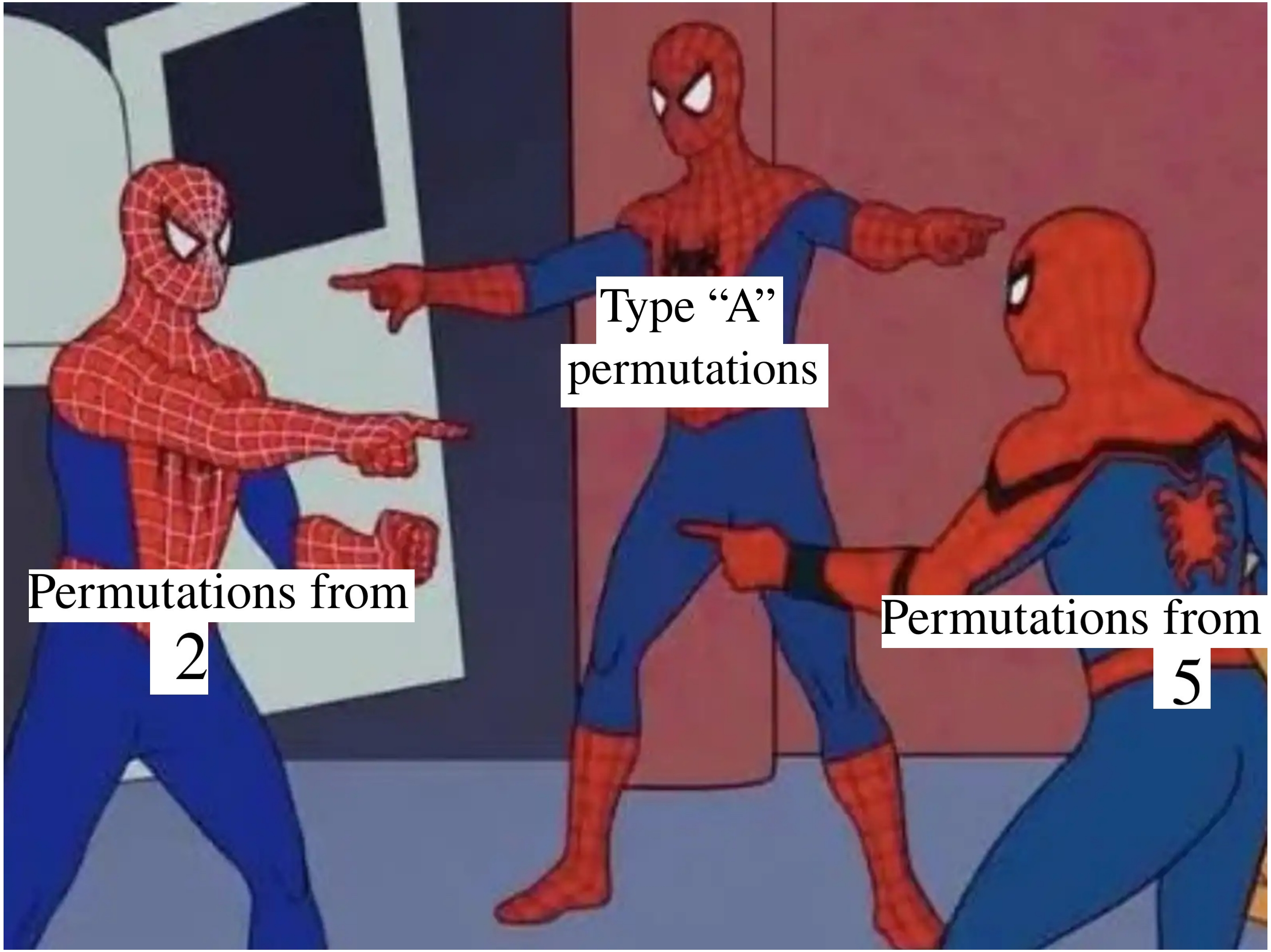


Figure: Permutations from[2]

Comparisons



[5] Oliver Coy Puente and Reynier A. de la Cruz Jiménez. On the Bit-Slice representations of some nonlinear bijective transformations. CTCrypt’23, 2023.

Known results about computational work in the LUT-6 basis

It is known[6] that:

- Random permutation $S\colon \mathbb{F}_2^8 \to \mathbb{F}_2^8$ can be implemented using 40 LUTs (about 812 GE)
- Permutations of “A” class can be implemented using only 19 LUTs (about 147 GE)

Class “A”, “B” and “C” permutations are effectively implemented by the FPGA.

[6] D.B. Fomin, D.I. Trifonov. Hardware implementation of one class of 8-bit permutations Prikladnaya Diskretnaya Matematika. Supplement, 134-137

Computational work fore some functions over the field $\mathbb{F}_{(2^2)^2}$ I

Let «·» — be the multiplication operation in the field $\mathbb{F}_{(2^2)^2}$, defined by an irreducible polynomial $g(x) = x^2 + x + \alpha$ in the polynomial basis $\{1, \beta\}$.

Then for $x, y \in \mathbb{F}_{(2^2)^2}$ $C_\Omega\left(x \cdot y; \mathbb{F}_{(2^2)^2}; \mathrm{Poly}\right) \leq 30$, $D_\Omega\left(x \cdot y; \mathbb{F}_{(2^2)^2}; \mathrm{Poly}\right) \leq 5$.

For $x, y \in \mathbb{F}_{(2^2)^2}$ $C_\Omega\left(x \cdot y; \mathbb{F}_{(2^2)^2}; \mathrm{PtN}\right) \leq 30$, $D_\Omega\left(x \cdot y; \mathbb{F}_{(2^2)^2}; \mathrm{PtN}\right) \leq 5$.

Computational work fore some functions over the field $\mathbb{F}_{(2^2)^2}$ II

Metric:	C			D		
Basis:	Poly	Norm	NtP	Poly	Norm	NtP
x^2	≤ 3	≤ 4	≤ 4	≤ 2	≤ 2	≤ 2
x^4	≤ 2	0	≤ 2	1	0	1
x^7	≤ 29	≤ 29	≤ 29	≤ 8	≤ 7	≤ 7
x^8	≤ 3	≤ 4	≤ 4	≤ 2	≤ 2	≤ 2
x^{11}	≤ 26	≤ 26	≤ 26	≤ 8	≤ 7	≤ 7
x^{13}	≤ 29	≤ 29	≤ 29	≤ 8	≤ 7	≤ 7
x^{14}	≤ 26	≤ 26	≤ 26	≤ 8	≤ 7	≤ 7
$x^7 + x^4 + x$	≤ 31	≤ 29	≤ 29	≤ 9	≤ 7	≤ 7

Computational work for MUX I

According to the definition of the parametric families of types “A”, “B” and “G” a calculation similar to the following occurs:
«$\texttt{if}\ x_1 = 0\ \texttt{then}\ y = \widehat{\pi}(x_0)\ \texttt{else}\ y = \pi_2(\pi_0(x_0) \cdot \pi_1(x_1))$»,
where $\pi_0, \pi_1, \pi_2, \widehat{\pi}$ are bijective permutations of $\mathbb{F}_2^4$.

The combinational complexity of $F(x_0, x_1) = \mathrm{Ind}_0(x_1) \cdot \widehat{\pi}(x_0) + \pi_2(\pi_0(x_0) \cdot \pi_1(x_1))$ is estimated by:

$$C_\Omega\left(\widehat{\pi}\right) + C_\Omega\left(\pi_2(\pi_0(x_0) \cdot \pi_1(x_1))\right) + 12.$$

The depth of the function that implements the above formula is equal:

$$\max\left\{4, D_\Omega\left(\widehat{\pi}\right) + 2, D_\Omega\left(\pi_2(\pi_0(x_0) \cdot \pi_1(x_1))\right) + 1\right\}.$$

Computational work estimation for some parametric families of permutations I

To implement a permutation from the parametric family of type “G” it is necessary to implement two functions, each of which consists of three permutations (two of them monomial), an operation of multiplication and a multiplexer. The combinational complexity of such a permutation is estimated by the following value:

$$C_\Omega(x^\alpha) + C_\Omega(x^\beta) + C_\Omega(x^\gamma) + C_\Omega(x^\delta) + 2C_\Omega(\cdot) + C_\Omega\left(\widehat{\pi}_1\right) + C_\Omega\left(\widehat{\pi}_2\right) + 2C_\Omega(\mathrm{MUX}) \leq$$
$$\leq 4 \cdot C_\Omega(x^7) + 2C_\Omega(\cdot) + 2C_\Omega(x^7 + x^4 + x) + 2 \cdot 30 + 2 \cdot 12 = 322.$$

Computational work estimation for some parametric families of permutations II

This value can be greatly overestimated. Consider the following permutation $S(x_1, x_2) = (y_1, y_2)$:

$$y_1 = \begin{cases} x_1 \cdot x_2^2, & x_2 \neq 0 \\ x_1^{-1}, & x_2 = 0 \end{cases},$$

$$y_2 = \begin{cases} x_1^{-1} \cdot x_2^{-1}, & x_1 \neq 0 \\ x_2^{-1}, & x_1 = 0 \end{cases}.$$

Its combinational complexity in the considered basis obviously does not exceed 147.

At the same depth, the permutation G defined earlier has combinational complexity equal to 144.

Computational work estimation for some parametric families of permutations III

Let us estimate the depth of the formula specifying a permutation from a parametric family of type “G”:

$$\max\left\{4, D_{\Omega}\left(\widehat{\pi}_1\right)+2, D_{\Omega}\left(\widehat{\pi}_2\right)+2, D_{\Omega}\left(x_1^{\alpha}\cdot x_2^{\beta}\right)+1, D_{\Omega}\left(x_1^{\gamma}\cdot x_2^{\delta}\right)+1\right\}\leq$$
$$\leq\max\left\{4, 8+6+2, \max\left\{D_{\Omega}\left(x_1^{\alpha}\right), D_{\Omega}\left(x_2^{\beta}\right), D_{\Omega}\left(x_1^{\gamma}\right), D_{\Omega}\left(x_2^{\delta}\right)\right\}+6\right\}\leq$$
$$\leq\max\left\{4, 8+6+2, 8+6\right\}\leq 16.$$

sboxgates

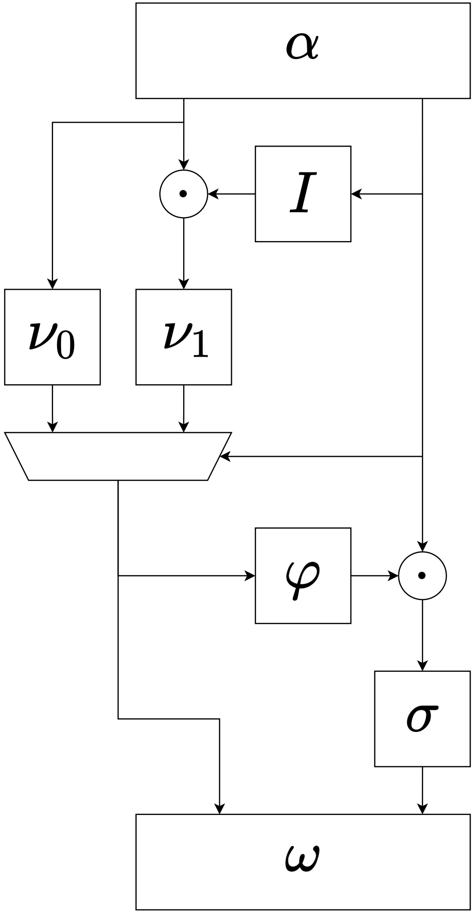
sboxgates[7] is a very powerful tool for finding bitclise implementations of functions $\mathbb{F}_2^n \to \mathbb{F}_2^m$, $n, m \leq 8$

[7] Marcus Dansarie. sboxgates: A program for finding low gate count implementations of S-boxes. Journal of Open Source Software vol.6:62, pages 2946, 2021

Computational work estimation for Kuznyechik using `sboxgates`

Today new bitslice representation of Kuznyechik s-box have been presented by Oliver Coy Puente and Reynier A. de la Cruz Jiménez:

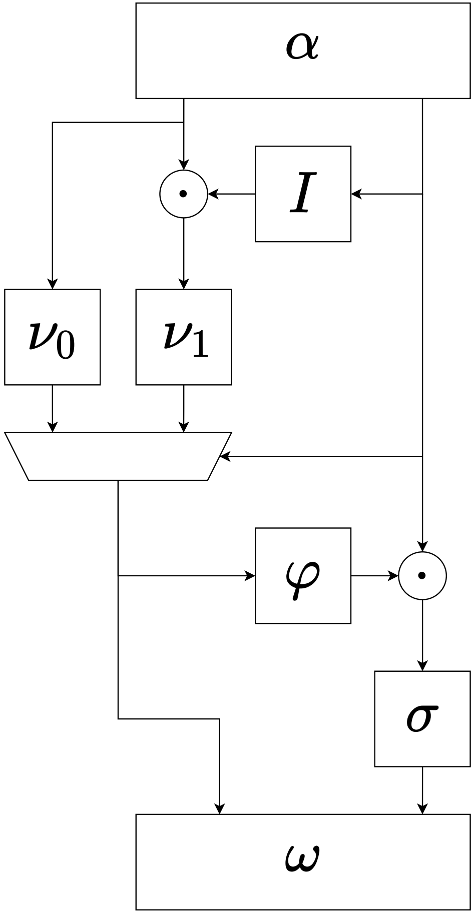
Operation	C
α	9
ω	5
I	20
ν_0	19
ν_1	12
ϕ	18
σ	19
MUX	15
$\odot$	31
Total:	179



Computational work estimation for Kuznyechik using `sboxgates`

`sboxgates` realizes a heuristic search algorithm:

Operation	C
α	9
ω	5
I	20
ν_0	19
ν_1	12
ϕ	18
σ	~~19~~ 17
MUX	15
$\odot$	31
Total:	~~179~~ 177

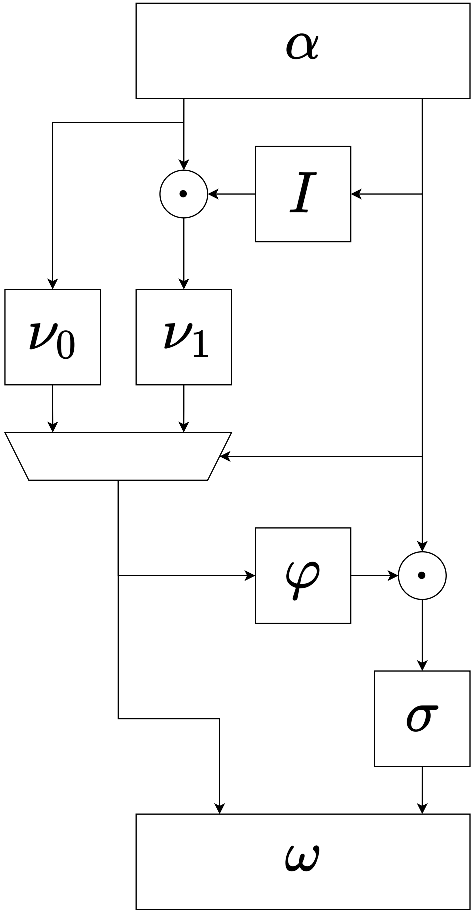


Computational work estimation for Kuznyechik using `sboxgates`

$$\mathrm{Ind}(r = 0) \cdot (\nu_0(l)) \oplus \overline{\mathrm{Ind}(r = 0)} \cdot \nu_1 (l \cdot I(r)) =$$
$$= \mathrm{Ind}(r = 0) \cdot (\nu_0(l) \oplus \nu_1(0))) \oplus \nu_1 (l \cdot I(r))$$

Let's denote $\nu_0(l) + \nu_1(a)$ as $\nu_0'(l)$:

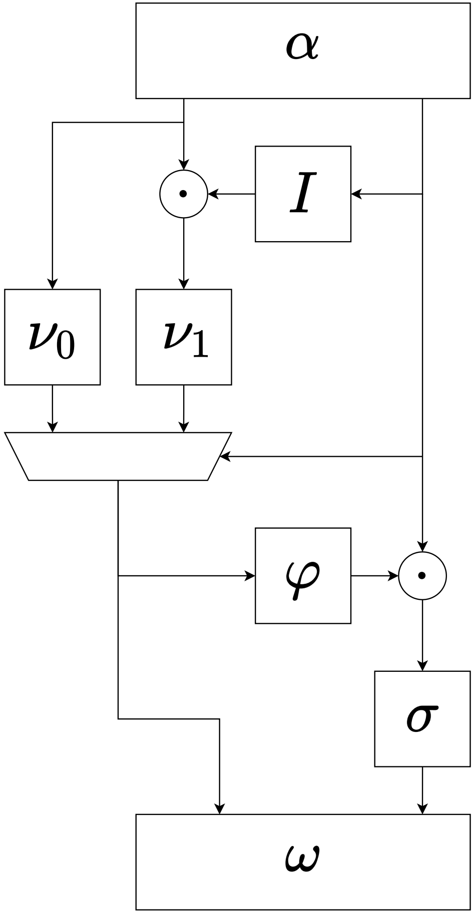
Operation	C
α	9
ω	5
I	20
ν_0'	~~19~~ 20
ν_1	12
ϕ	18
σ	17
MUX	~~15~~ 12
$\odot$	31
Total:	~~177~~ 175



Computational work estimation for Kuznyechik using sboxgates

Add NOT operator:

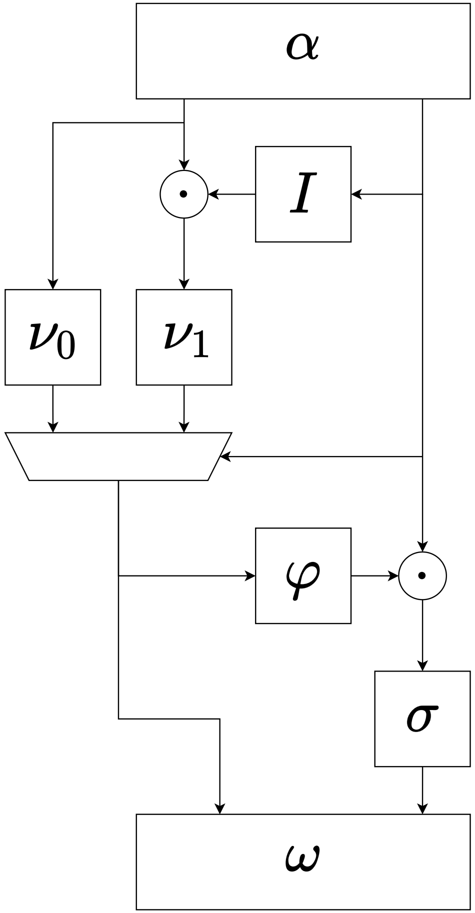
Operation	C
α	9
ω	5
I	~~20~~ 19
ν_0'	~~20~~ 19
ν_1	~~12~~ 11
ϕ	18
σ	17
MUX	12
$\odot$	31
Total:	~~175~~ 172



Computational work estimation for Kuznyechik using `sboxgates`

Result:

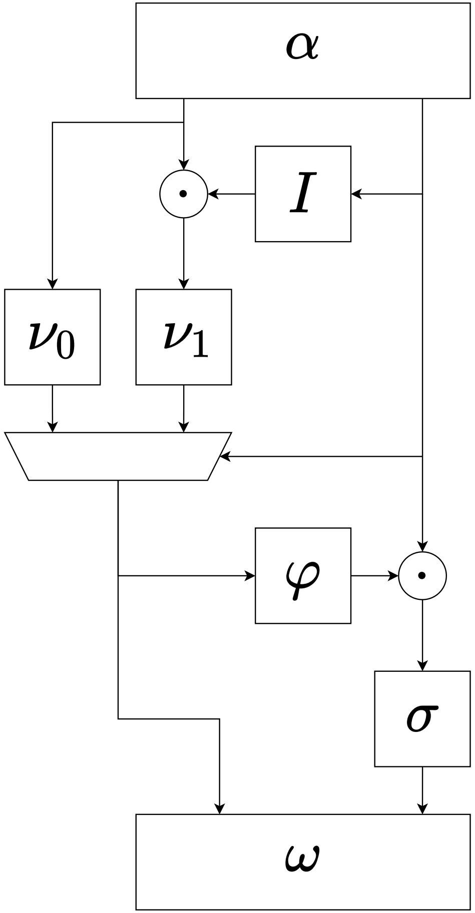
Operation	C
α	9
ω	5
I	19
ν_0'	19
ν_1	11
ϕ	18
σ	17
MUX	12
$\odot$	31
Total:	172



Computational work estimation for Kuznyechik using `sboxgates` and tower field math

There are 8 different linear transformations from $\mathbb{F}_{2^4}$ to $\mathbb{F}_{(2^2)^2}$ (using $x^2 + x + \alpha$ and $x^2 + x + \alpha^2$ irreducible polynomials for $\mathbb{F}_{2^2}$.)

As we know: $C_\Omega\left(x \cdot y; \mathbb{F}_{(2^2)^2}; \text{Poly}\right) \leq 30$ vs $C_\Omega\left(x \cdot y; \mathbb{F}_{2^4}; \text{Poly}\right) \leq 31$.

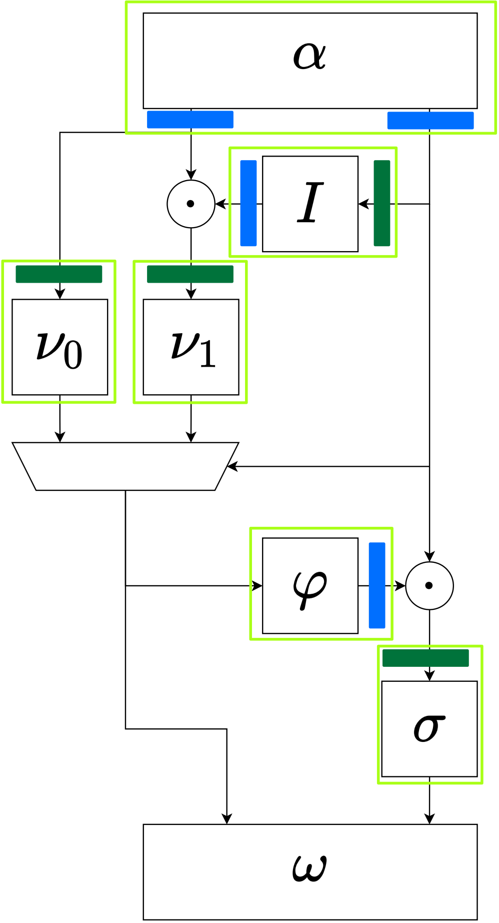


Computational work estimation for Kuznyechik using `sboxgates` and tower field math

Let $\boxed{si}$ be a linear permutation from polynomial representation of $\mathbb{F}_{2^4}$ to tower field representation $\mathbb{F}_{(2^2)^2}$ and $\boxed{sp} = si^{-1}$

Then we can consider new permutations:

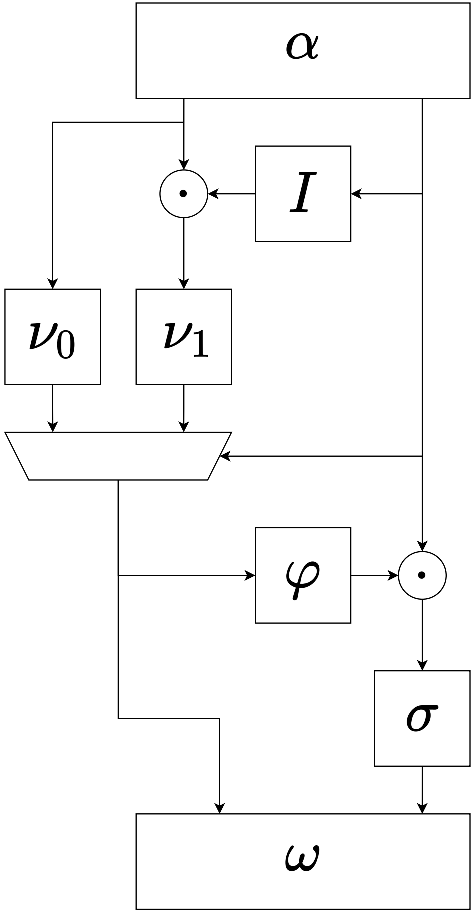
- $\widehat{\alpha}(x) = si[\alpha(x)\&\texttt{0x3}] \oplus (si[\alpha(x) \gg 3] \ll 3)$
- $\widehat{I} = si[I[sp[x]]]$
- $\widehat{\nu_0'}(x) = \nu_0'[sp[x]]$
- $\widehat{\nu_1}(x) = \nu_1[sp[x]]$
- $\widehat{\phi}(x) = si[\phi[x]]$
- $\widehat{\sigma}(x) = \sigma[sp[x]]$
- $\widehat{\omega}(x) = \omega(x)$



Computational work estimation for Kuznyechik using sboxgates and tower field math

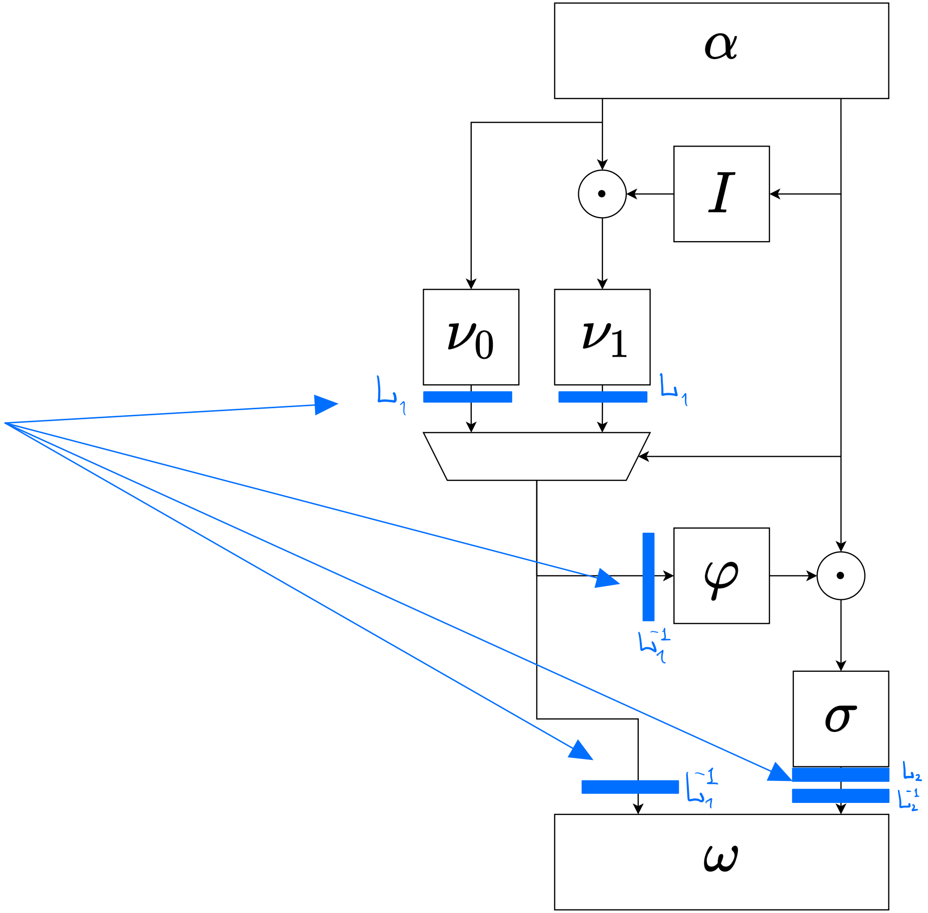
Result:

Operation	C	D
$\widehat{\alpha}$	11	5
$\widehat{\omega}$	5	2
$\widehat{I}$	16	10
$\widehat{\nu_0'}$	19	11
$\widehat{\nu_1}$	11	7
$\widehat{\phi}$	16	9
$\widehat{\sigma}$	19	12
MUX	12	2
$\odot$	30	5
Total:	169	55



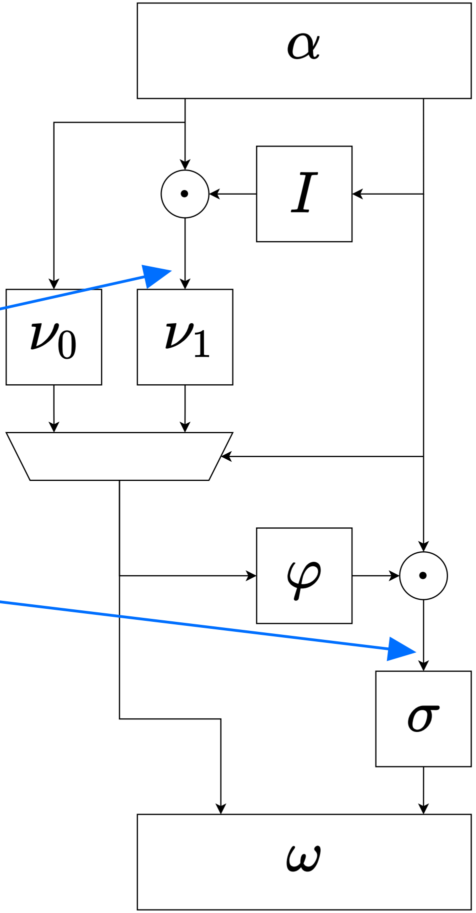
Other ways to optimizations

- using linear-equivalence permutations
- using mixed basis



Other ways to optimizations

- using linear-equivalence permutations
- using mixed basis



Computational work for one permutation

The permutation G defined earlier has combinational complexity equal to

$$2 \cdot 30 + 2 \cdot 12 + 2 \cdot 16 = 116.$$

The depth for this permutation is $7 + 2 + 5 = 14$.