

Cryptanalysis Results on Spook

Bringing Full Shadow-512 to the Light

Patrick Derbez¹, Paul Huynh², Virginie Lallemand², María Naya-Plasencia³, Léo Perrin³, André Schrottenloher³

¹ Université de Rennes, CNRS, Irista - Rennes, France

² Université de Lorraine, INRIA, Loria, CNRS - Nancy, France

³ INRIA - Paris, France

CRYPTO 2020 | August 18th, 2020 | your computer screen

Spook

Davide Bellizia, Francesco Berti, Olivier Bronchain, Gaëtan Cassiers, Sébastien Duval, Chun Guo, Gregor Leander, Gaëtan Leurent, Itamar Levi, Charles Momin, Olivier Pereira, Thomas Peters, François-Xavier Standaert, Balazs Udvarhelyi and Friedrich Wiemer

- 2nd round candidate to the NIST LWC standardization process
- Designed to achieve both resistance against side-channel analysis and low-energy implementations
- AEAD is provided using three sub-components
 - the Sponge One-Pass mode of operation (S1P)
 - the Clyde-128 tweakable block cipher
 - the [Shadow](#) permutation

Motivations

- Requirement for the permutation in the S1P mode of operation is that it provides **collision resistance** with respect to the 255 bits that generate the tag

“Hence, a more specific requirement is to prevent truncated differentials with probability larger than 2^{-128} for those 255 bits. A conservative heuristic for this purpose is to require that no differential characteristic has probability better than 2^{-385} , which happens after twelve rounds (six steps).”

- Mathematical cryptanalysis challenge proposed by the designers on the permutation

Summary of our work

- **Practical distinguishers** of the full 6-step version of the [Shadow-512](#) permutation and reduced 5-step version of [Shadow-384](#)
- **Practical forgeries** with [4-step Shadow](#) for the S1P mode of operation (nonce misuse scenario)

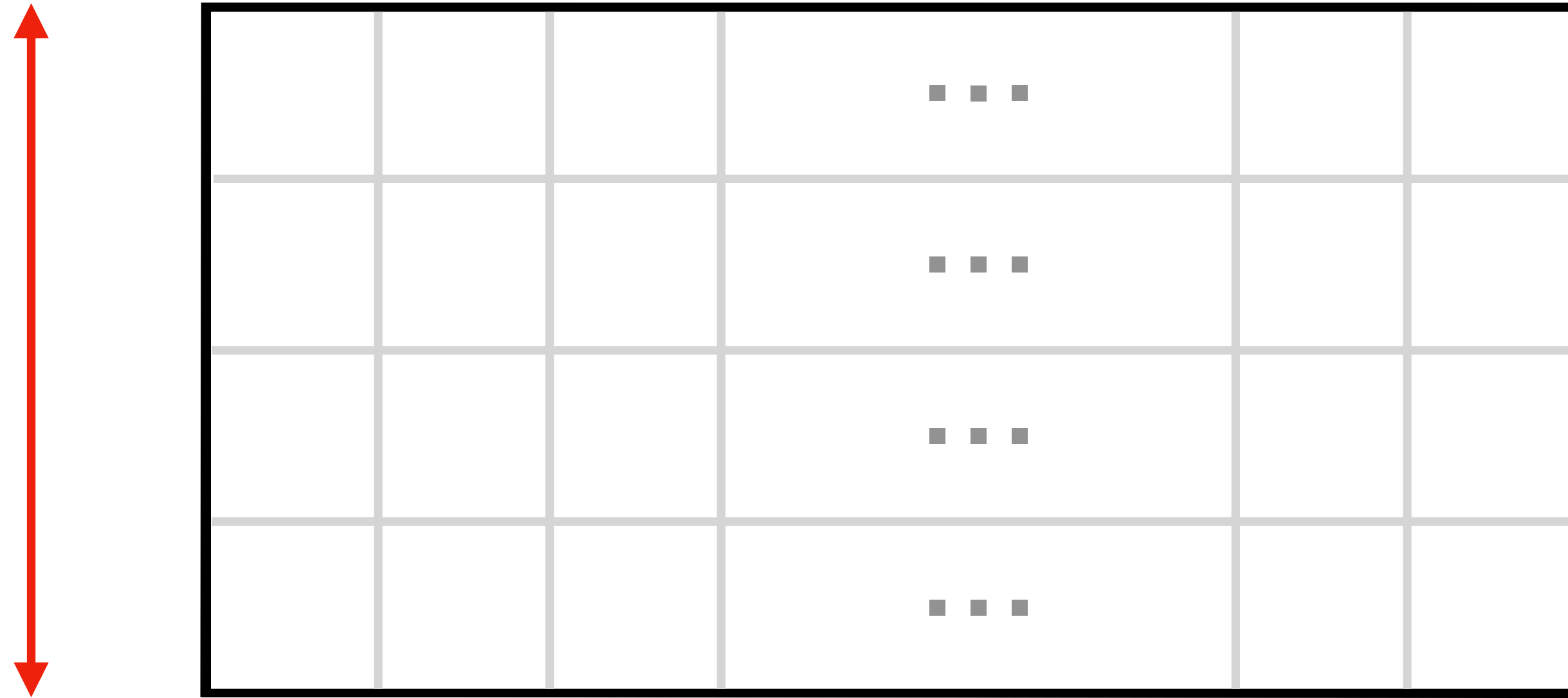
All the analyses are practical and have been implemented and tested. Source code available at:

<https://who.paris.inria.fr/Leo.Perrin/code/spook/index.html>

Description of Shadow

A Shadow **bundle**

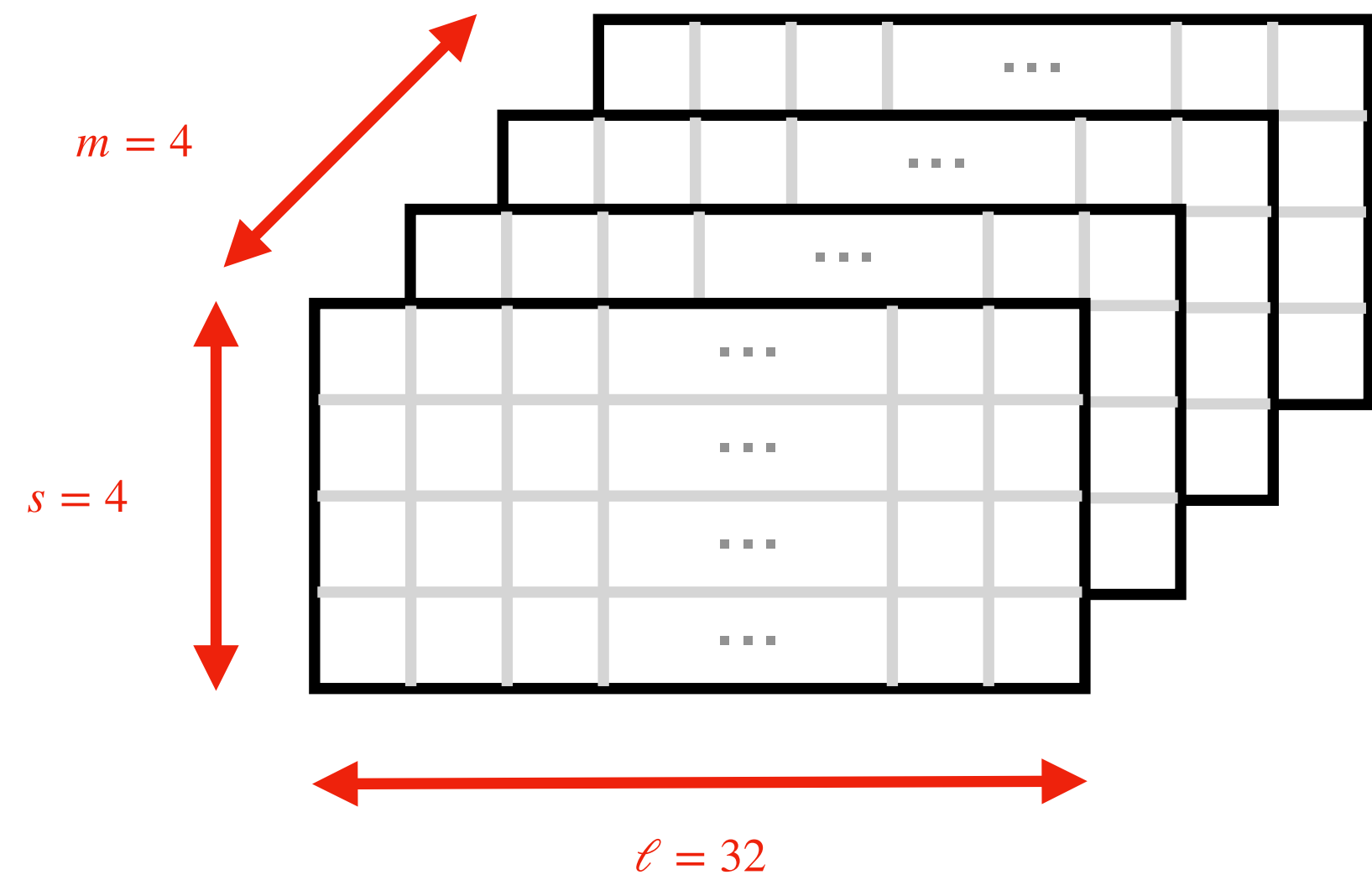
$s = 4$



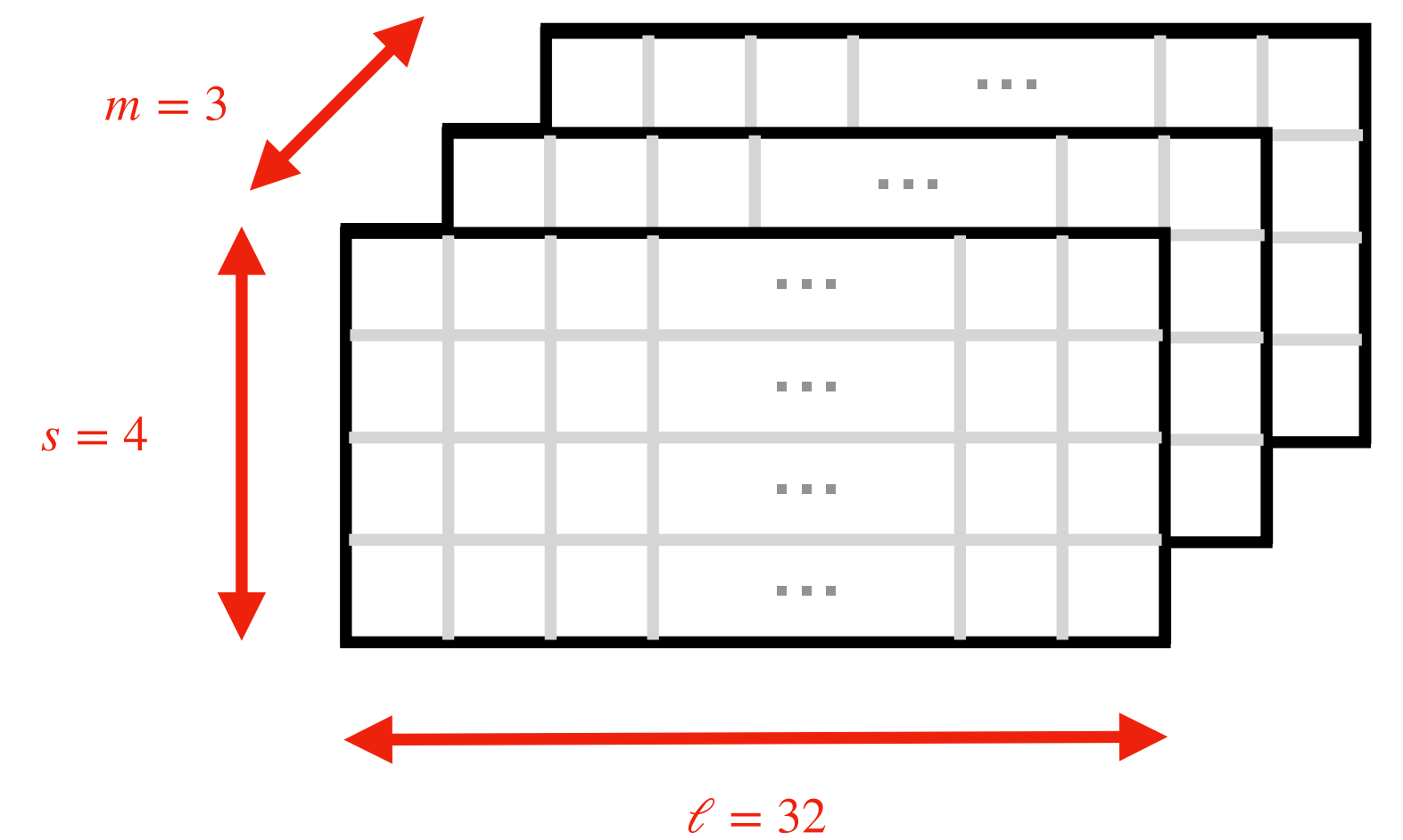
128 bits

$l = 32$

A Shadow **state**

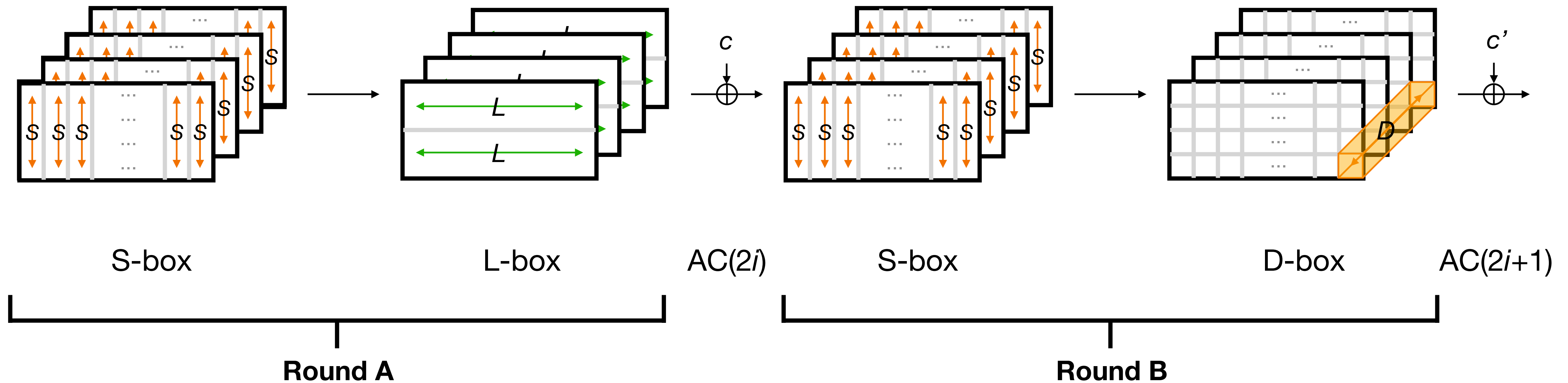


Shadow-512



Shadow-384

A Shadow encryption step



4-bit LFSR-generated constants added to **column i of bundle i**

6 steps to complete encryption

The D-layer

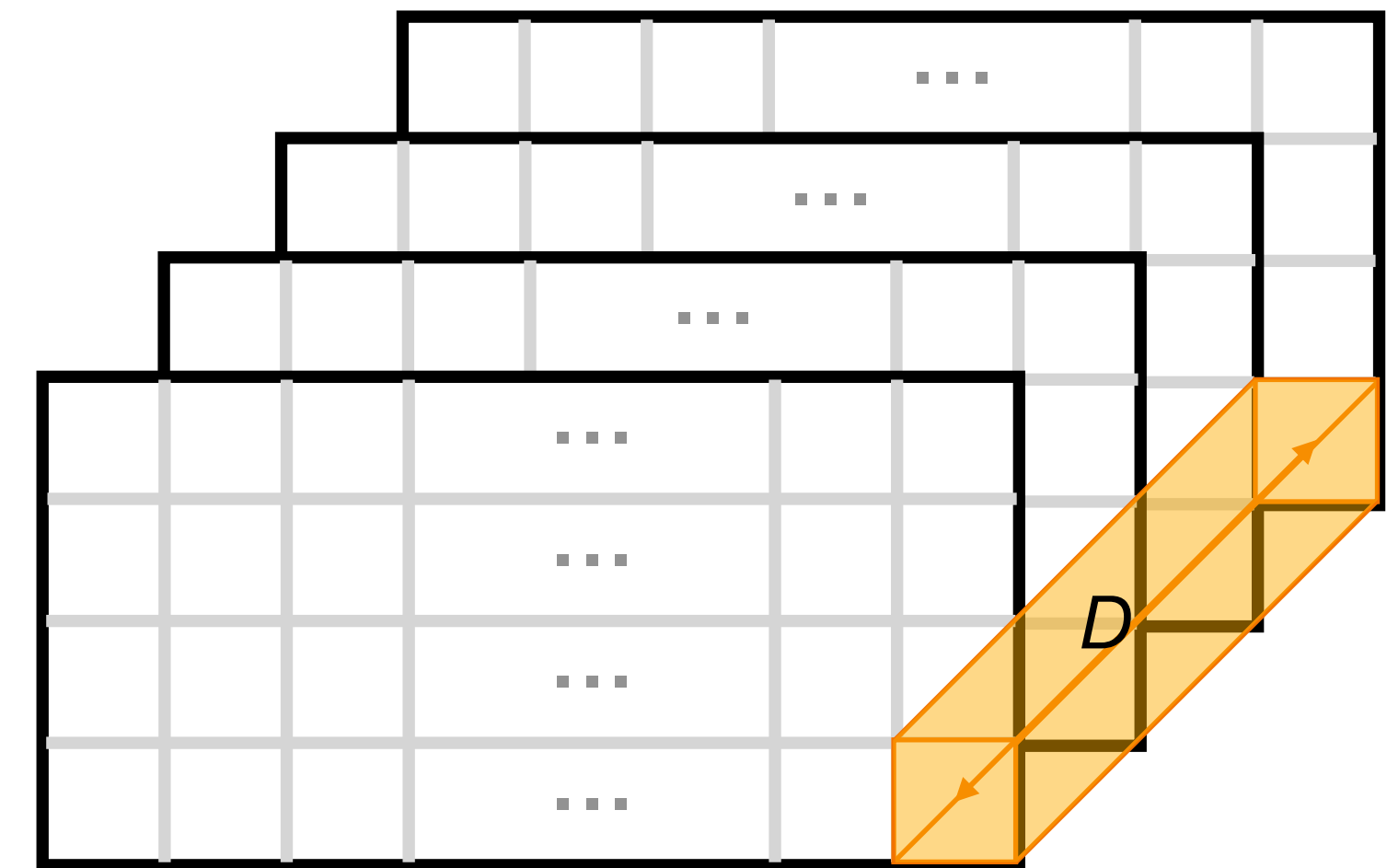
D is the only diffusion layer between the m bundles

○ Shadow-512:

$$D(a, b, c, d) = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix} \times \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}$$

○ Shadow-384:

$$D(a, b, c) = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \times \begin{pmatrix} a \\ b \\ c \end{pmatrix}$$



Main ideas

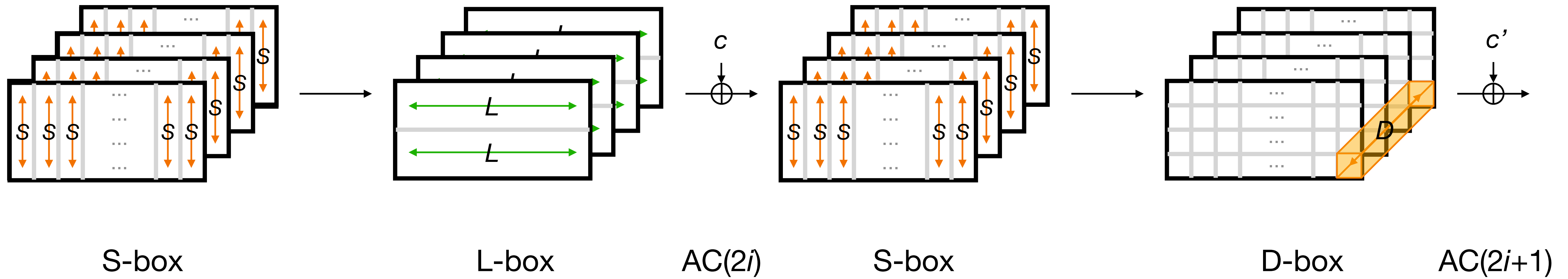
- Exploit the **similarity between the functions applied in parallel** on each bundle.
- **Truncated differential** distinguisher: variant of differentials in which only a portion of the difference is fixed while the remaining part is undetermined.

$$x \oplus x' = (*, *, *, 0) \text{ and } \text{shadow}(x) \oplus \text{shadow}(x') = D(0, 0, 0, *)$$

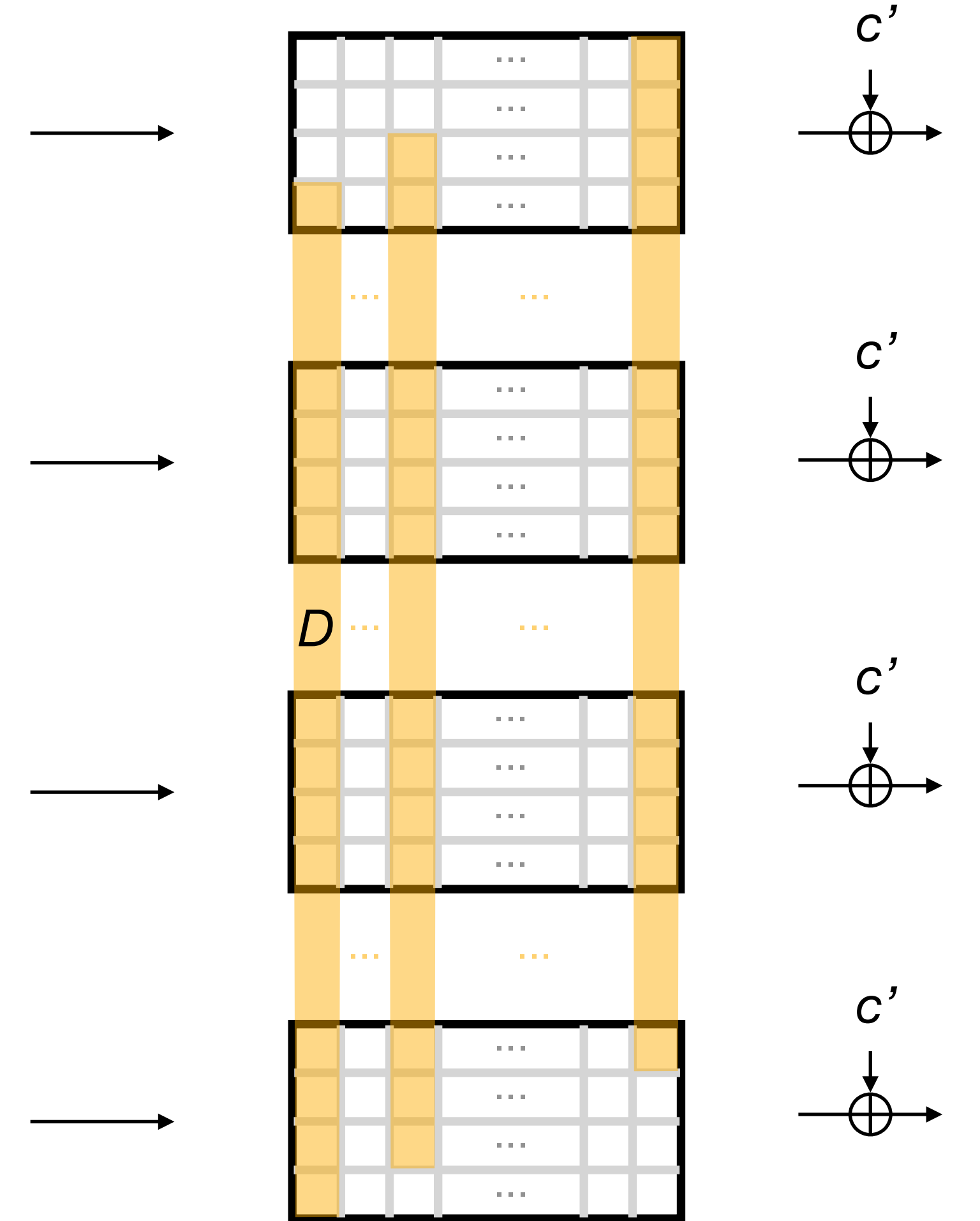
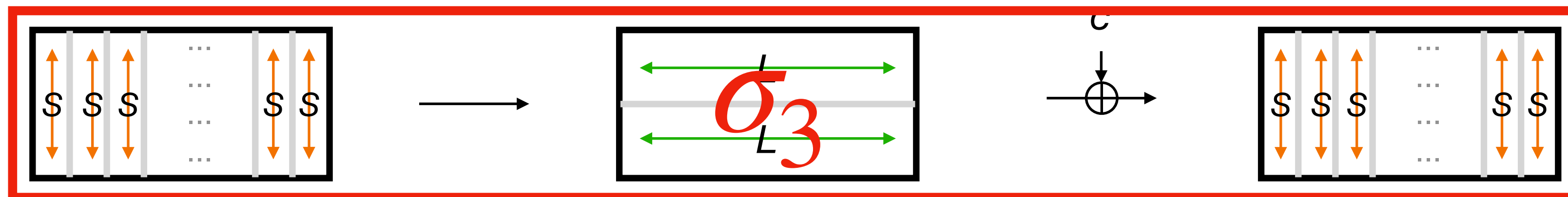
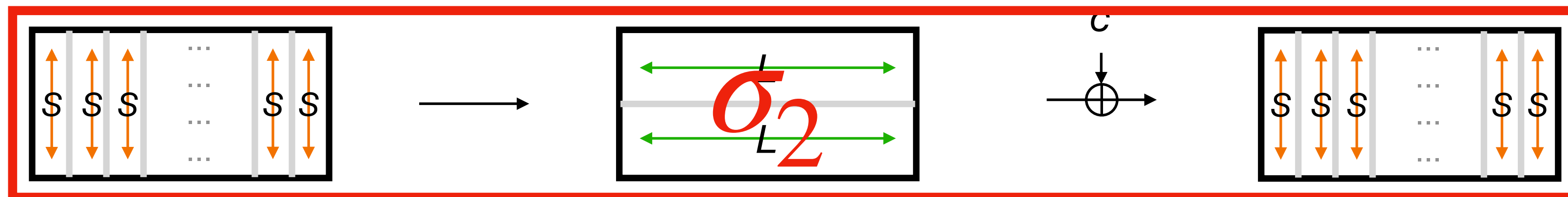
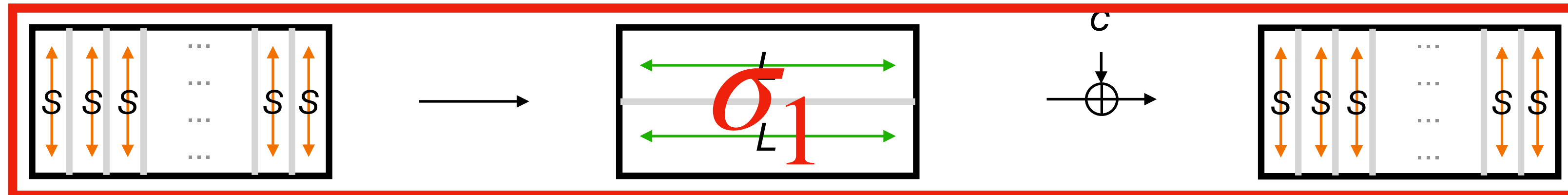
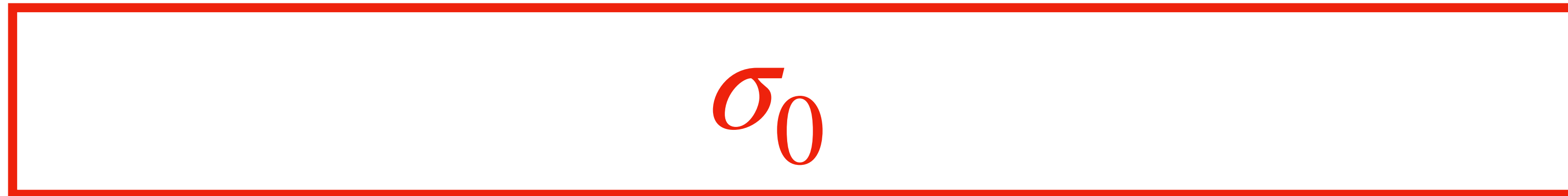
‘0’ the two bundles are identical

‘*’ the difference between the bundles is not determined

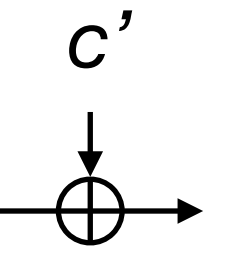
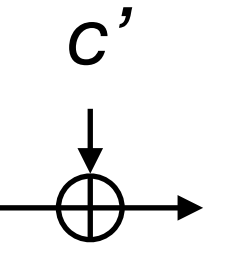
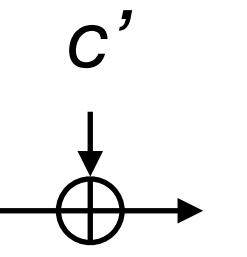
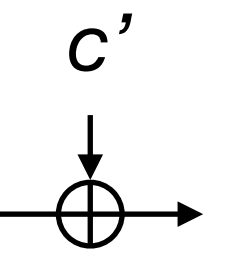
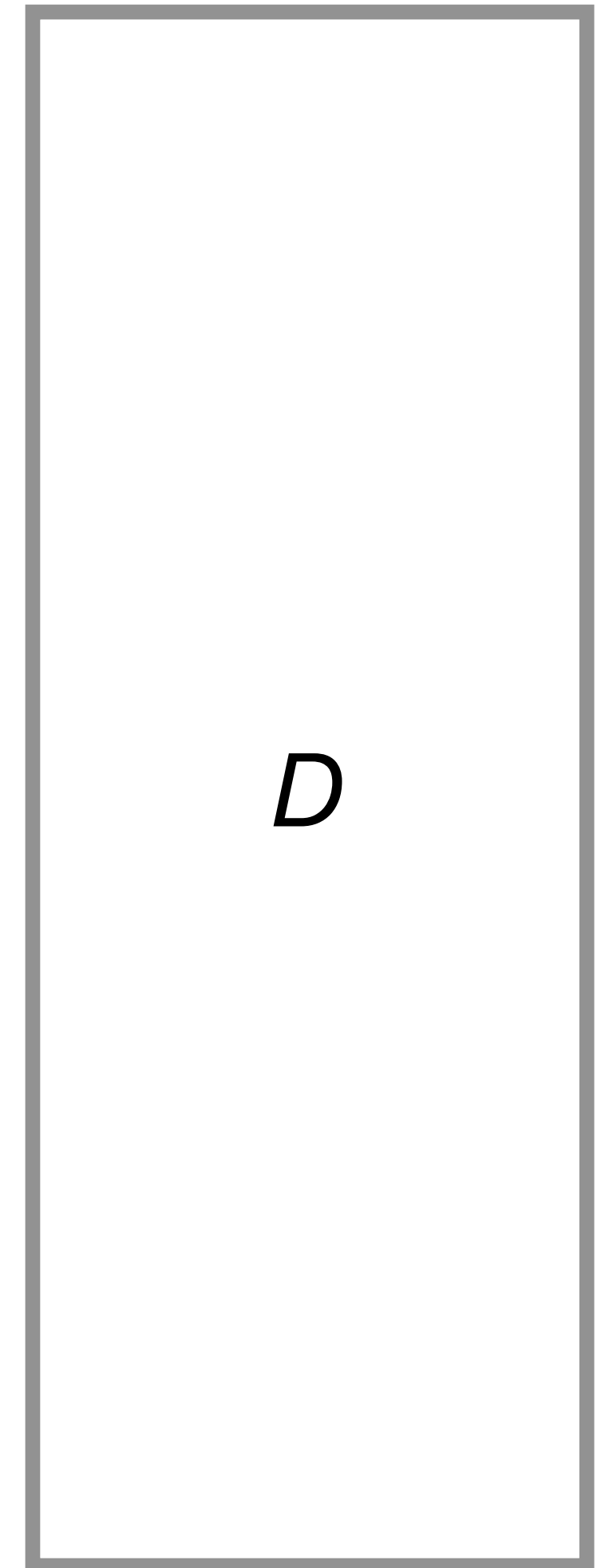
A Shadow step



A Shadow step **rewritten**

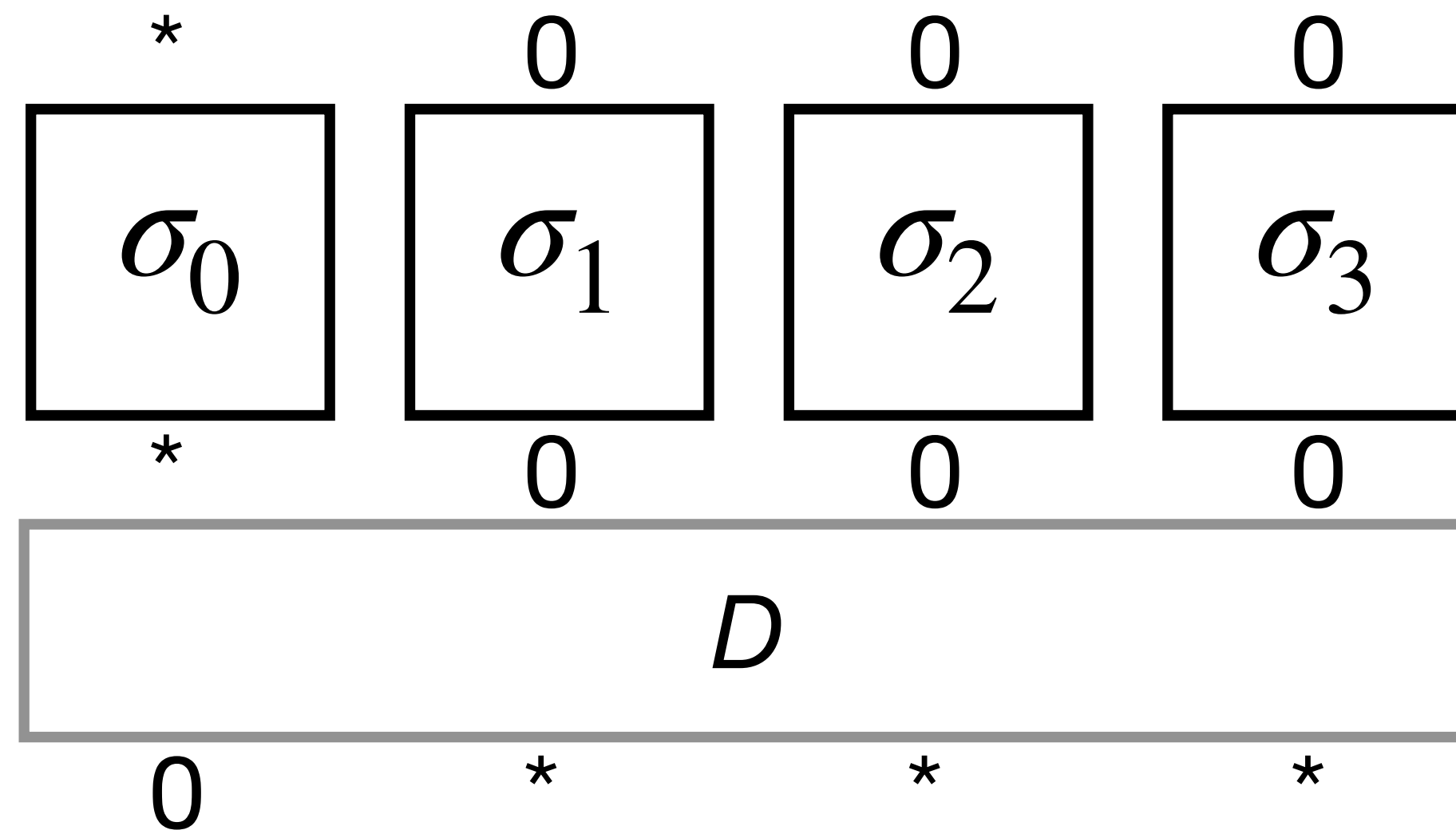


A Shadow step **rewritten**



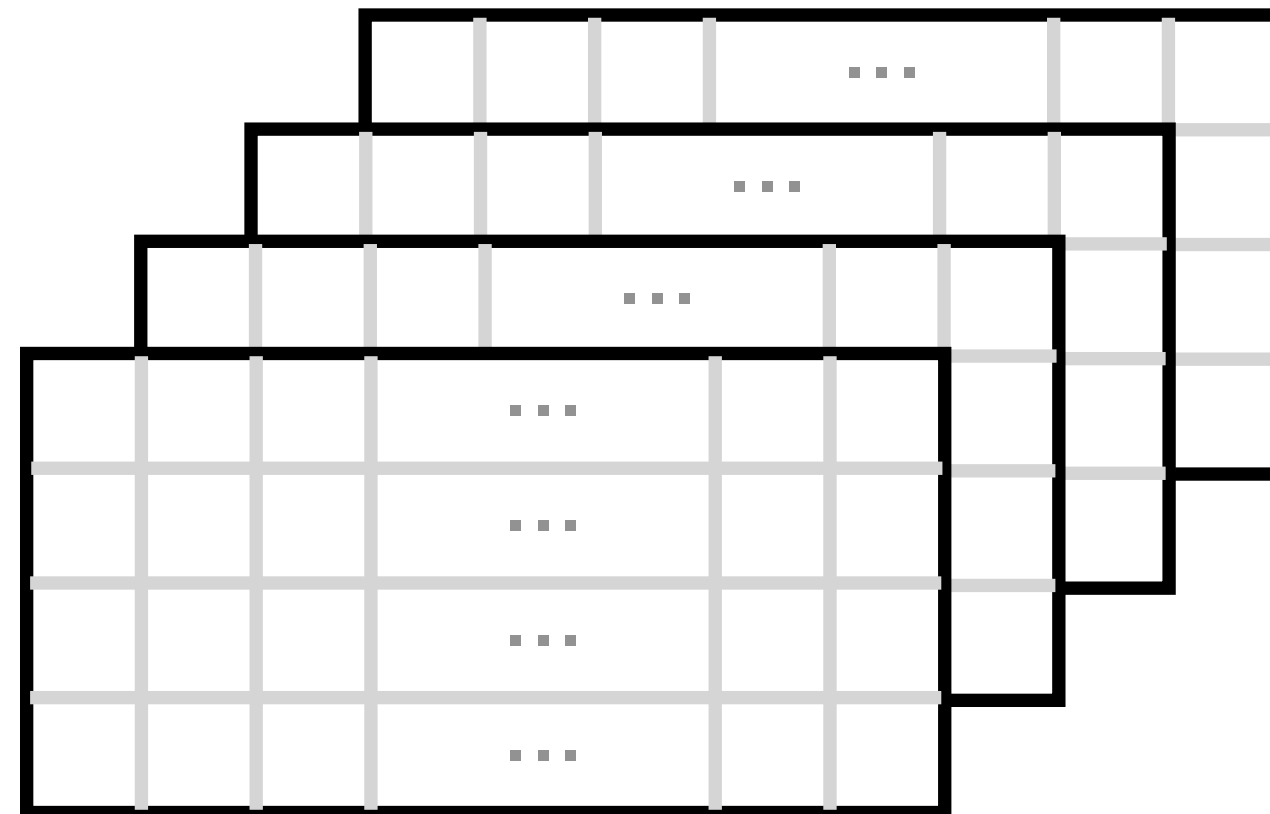
A Shadow step rewritten

Seen as an SPN, using four 128-bit **Super S-boxes** σ_i interleaved with a linear permutation D operating on the full state.



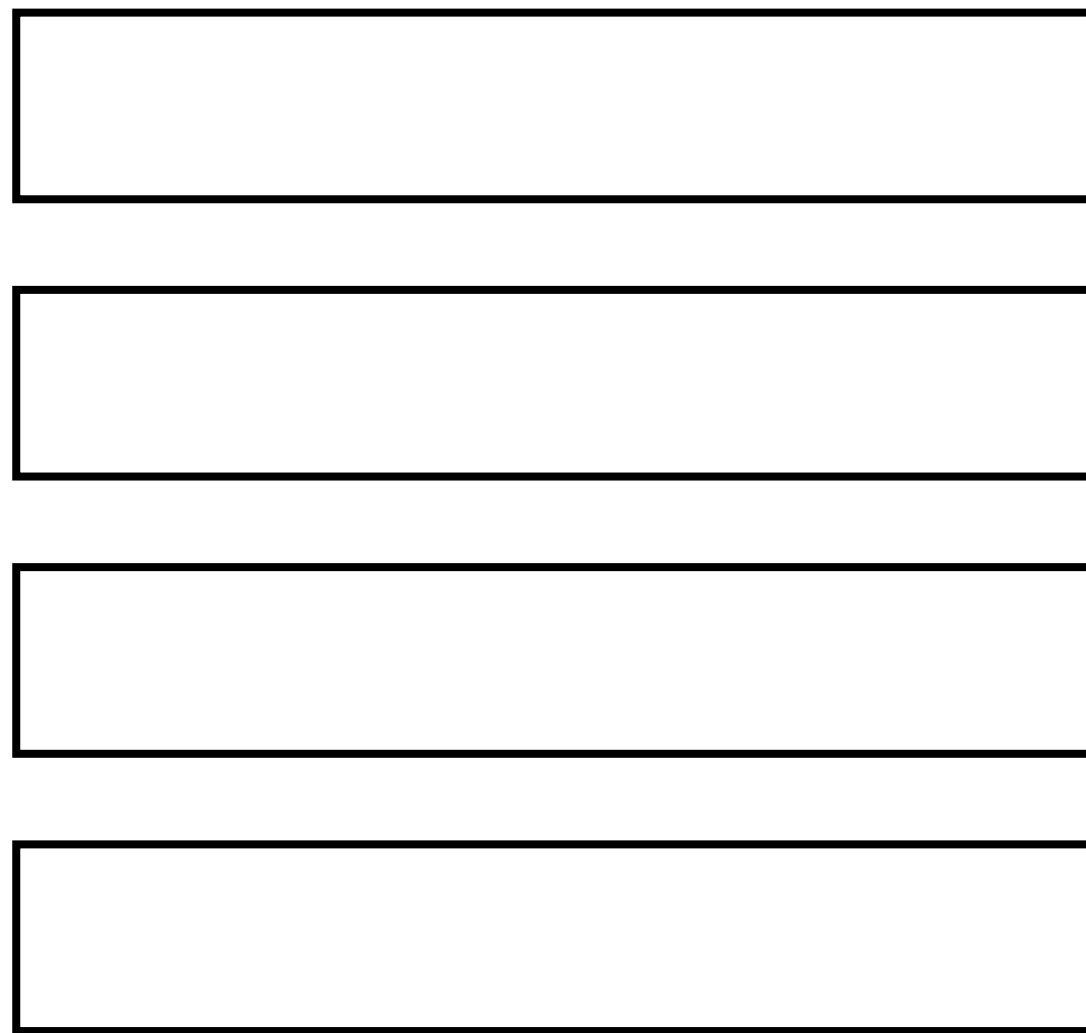
Structural observations

We call ***i*-identical** an internal state of Shadow in which i bundles are equal.



Structural observations

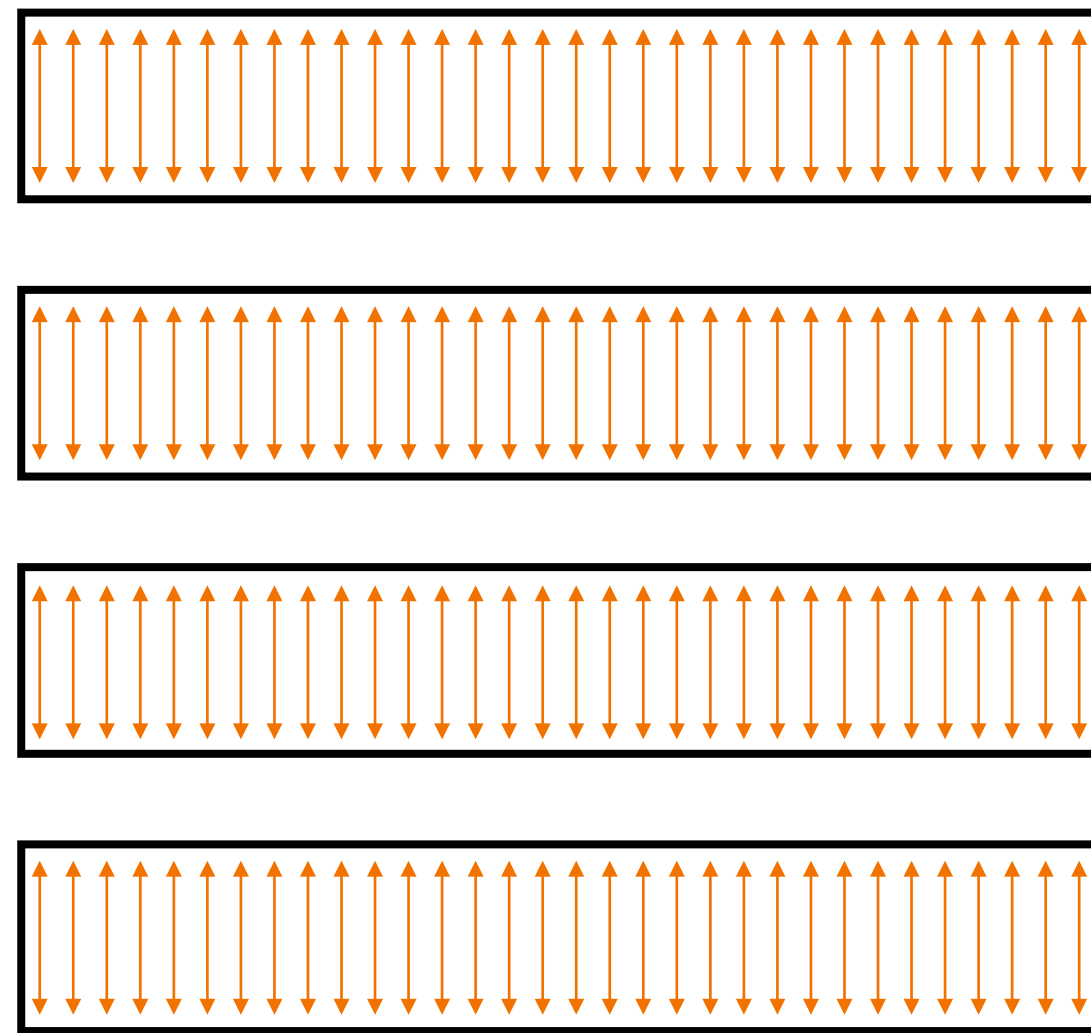
We call ***i*-identical** an internal state of Shadow in which i bundles are equal.



Initial state

Structural observations

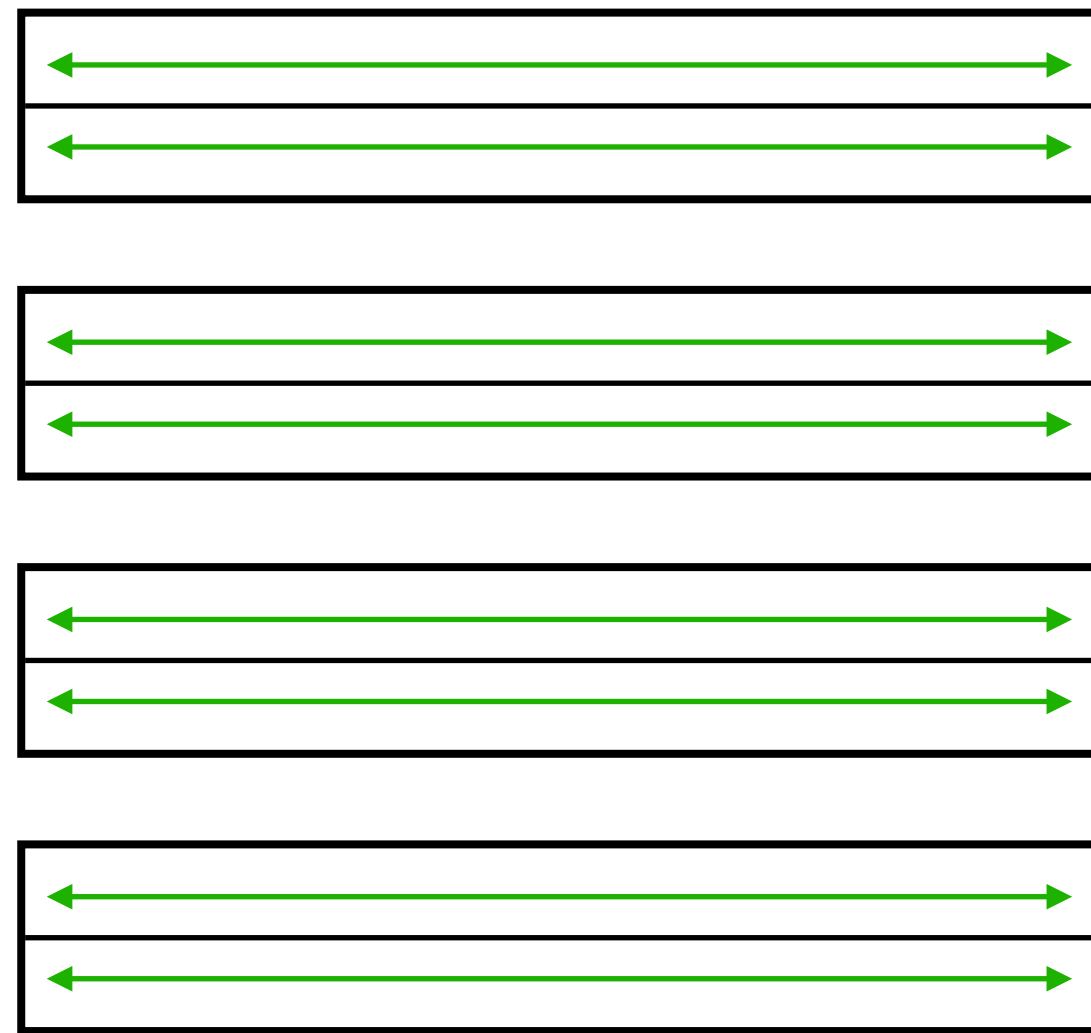
We call ***i*-identical** an internal state of Shadow in which *i* bundles are equal.



S-Box layer

Structural observations

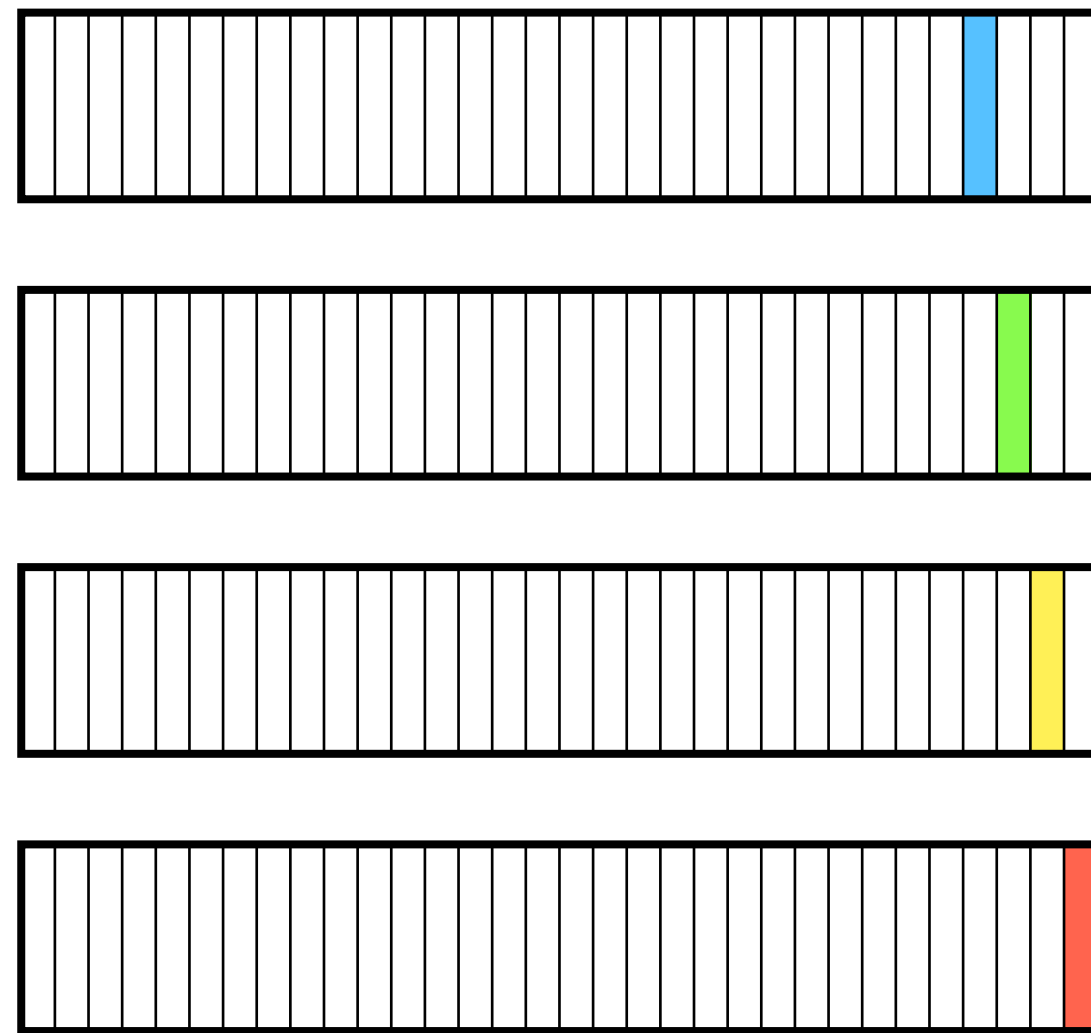
We call ***i*-identical** an internal state of Shadow in which *i* bundles are equal.



L-Box layer

Structural observations

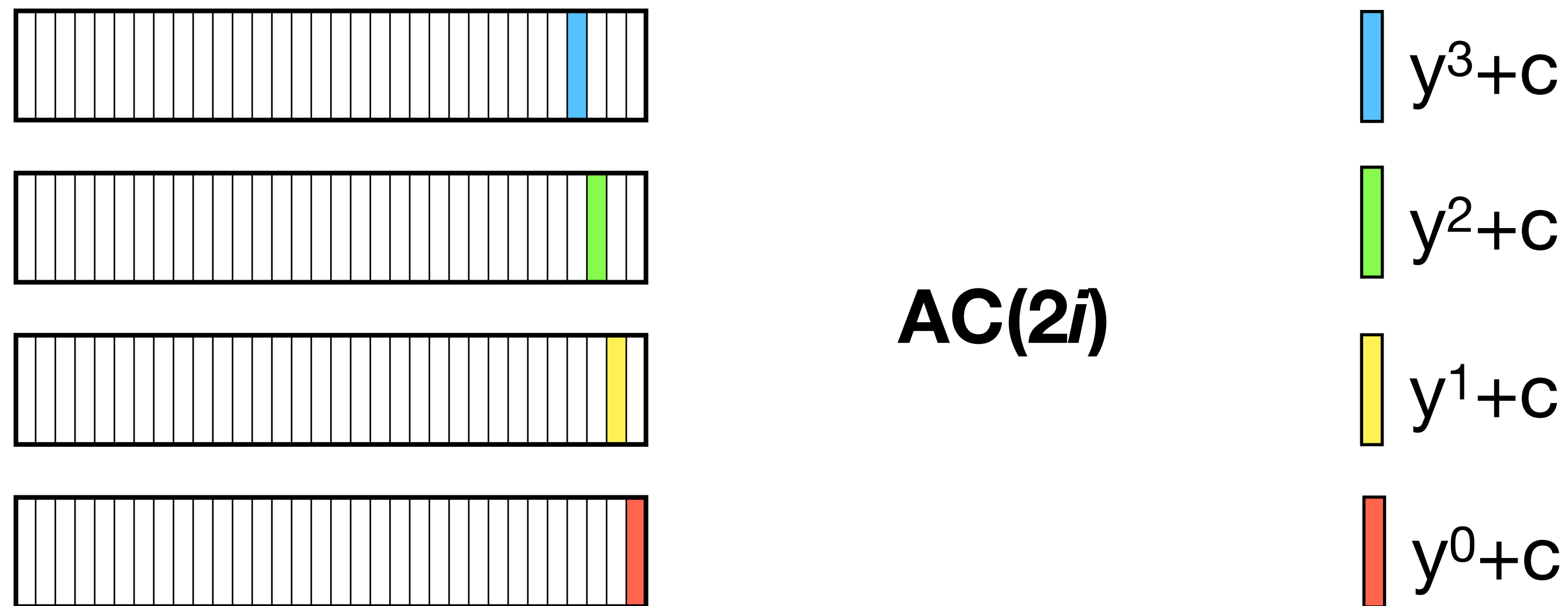
We call ***i*-identical** an internal state of Shadow in which *i* bundles are equal.



$AC(2i)$

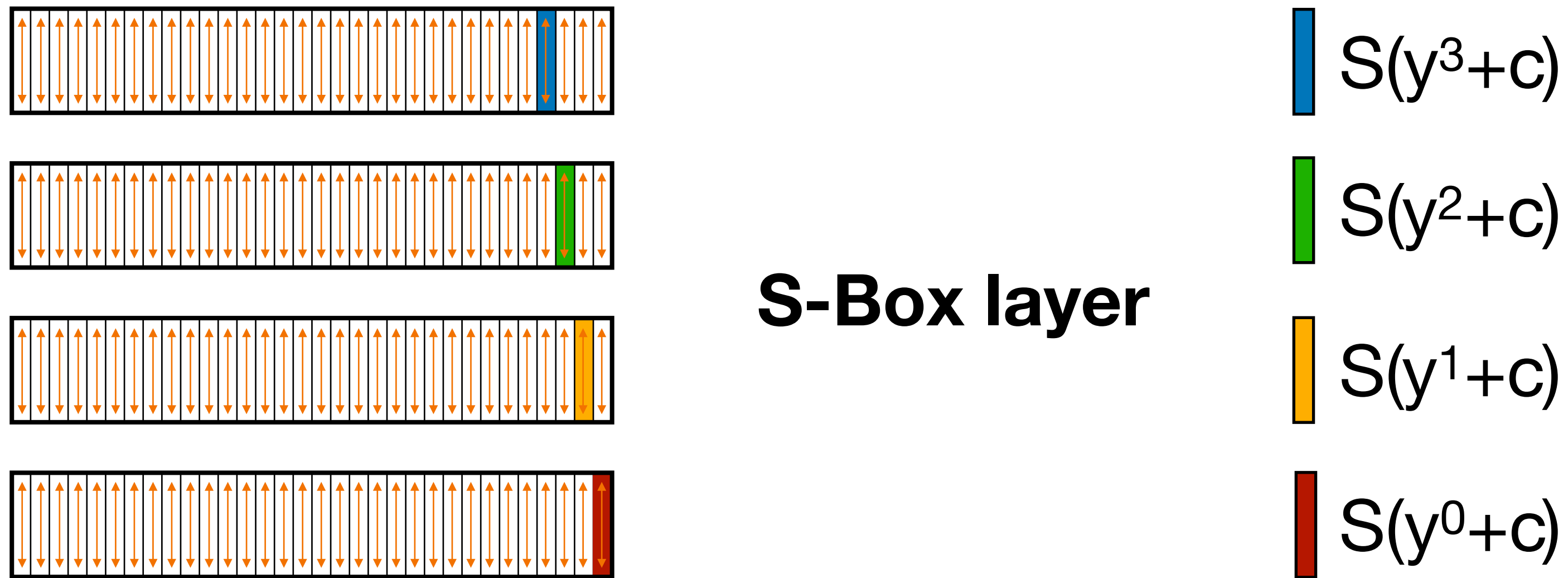
Structural observations

We call ***i*-identical** an internal state of Shadow in which *i* bundles are equal.



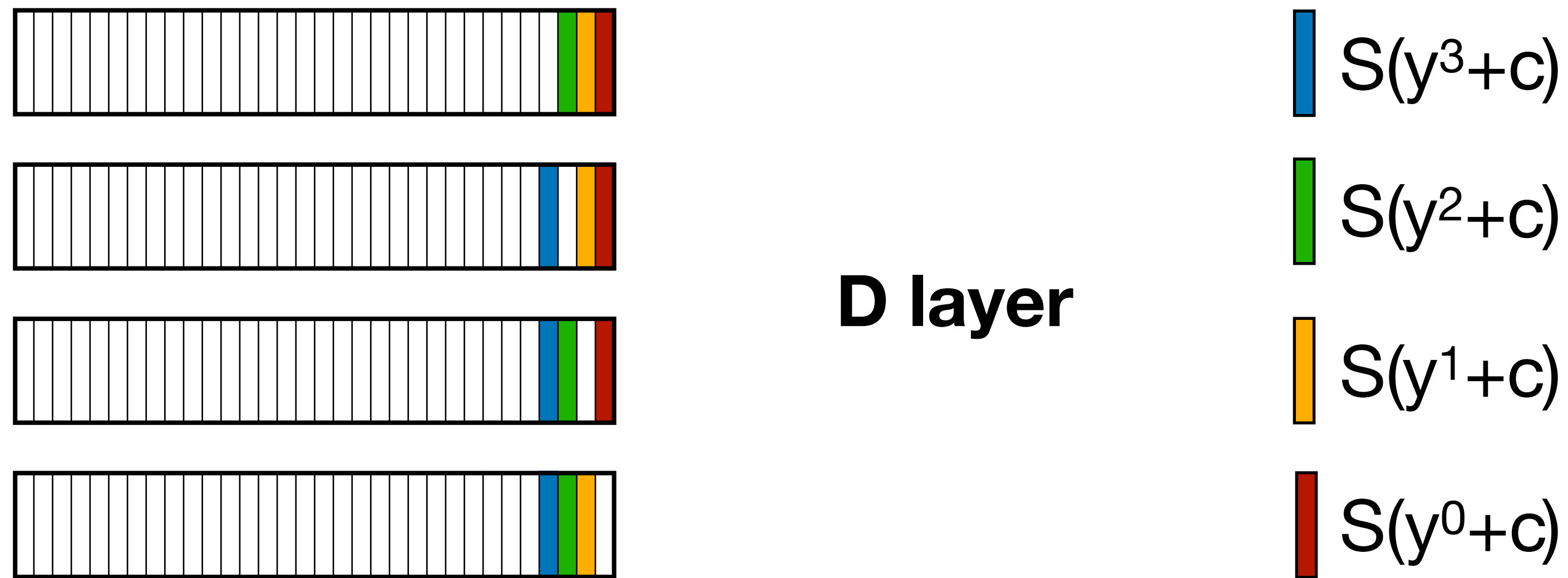
Structural observations

We call ***i*-identical** an internal state of Shadow in which *i* bundles are equal.



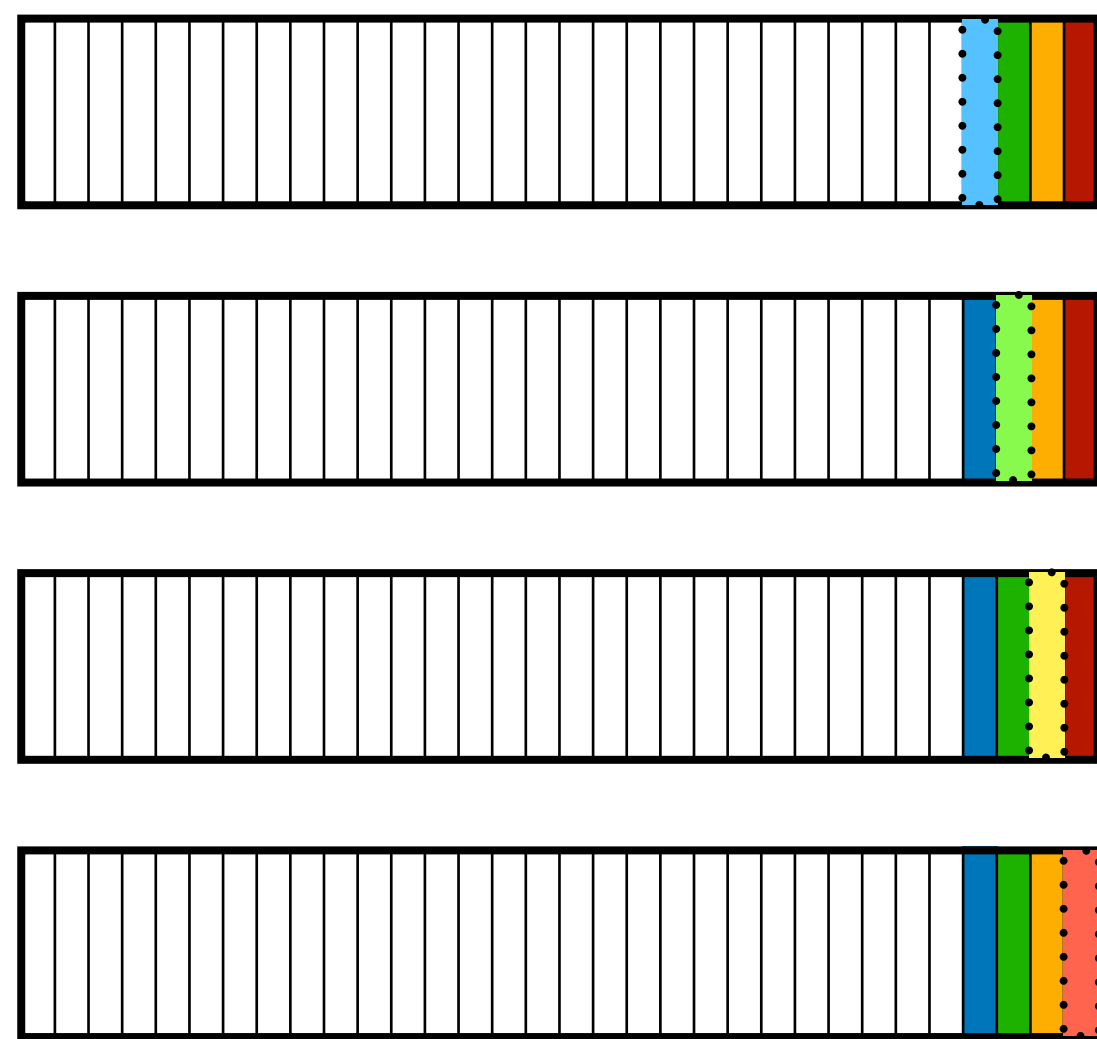
Structural observations

We call ***i*-identical** an internal state of Shadow in which *i* bundles are equal.

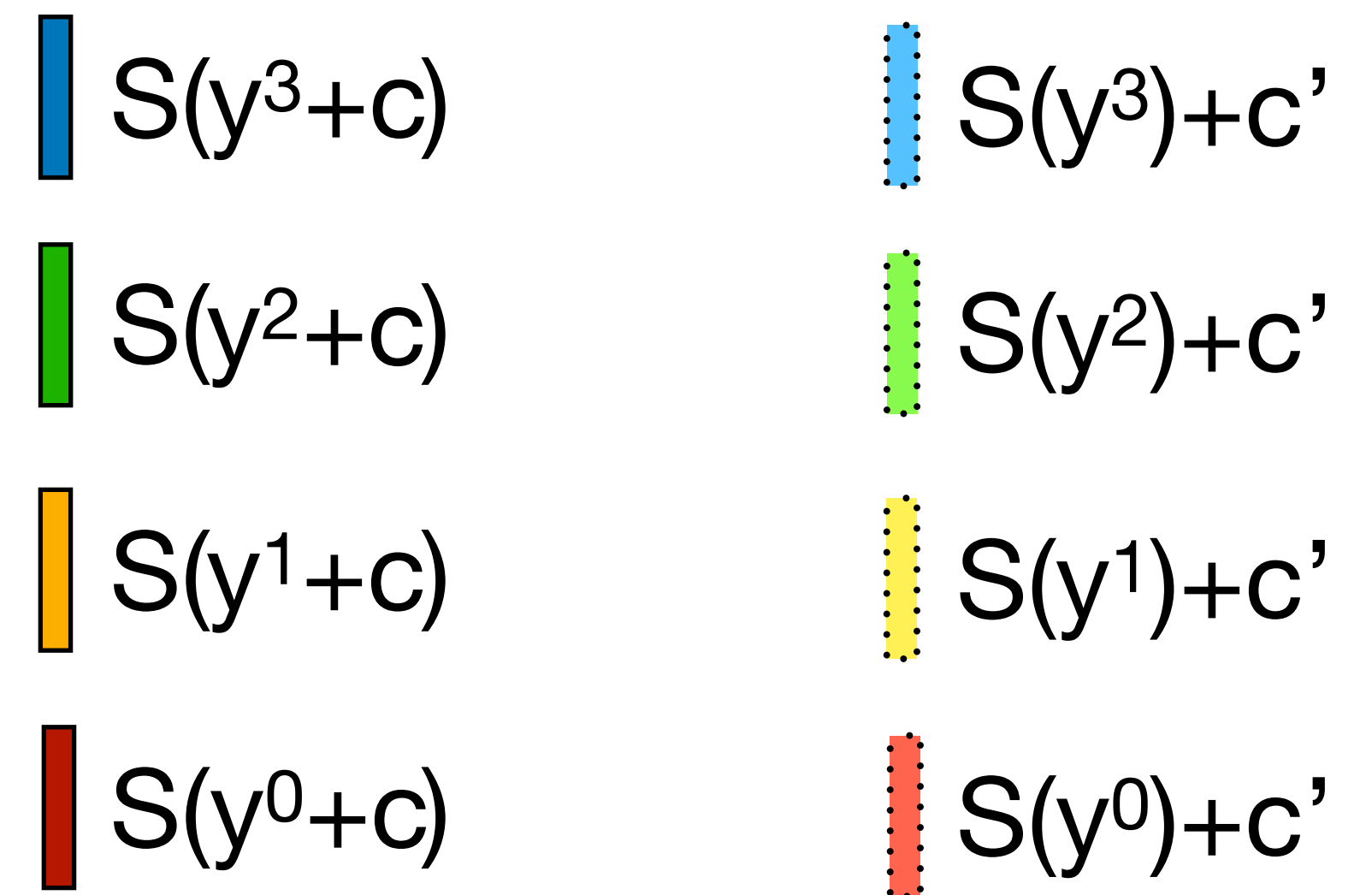


Structural observations

We call ***i*-identical** an internal state of Shadow in which *i* bundles are equal.

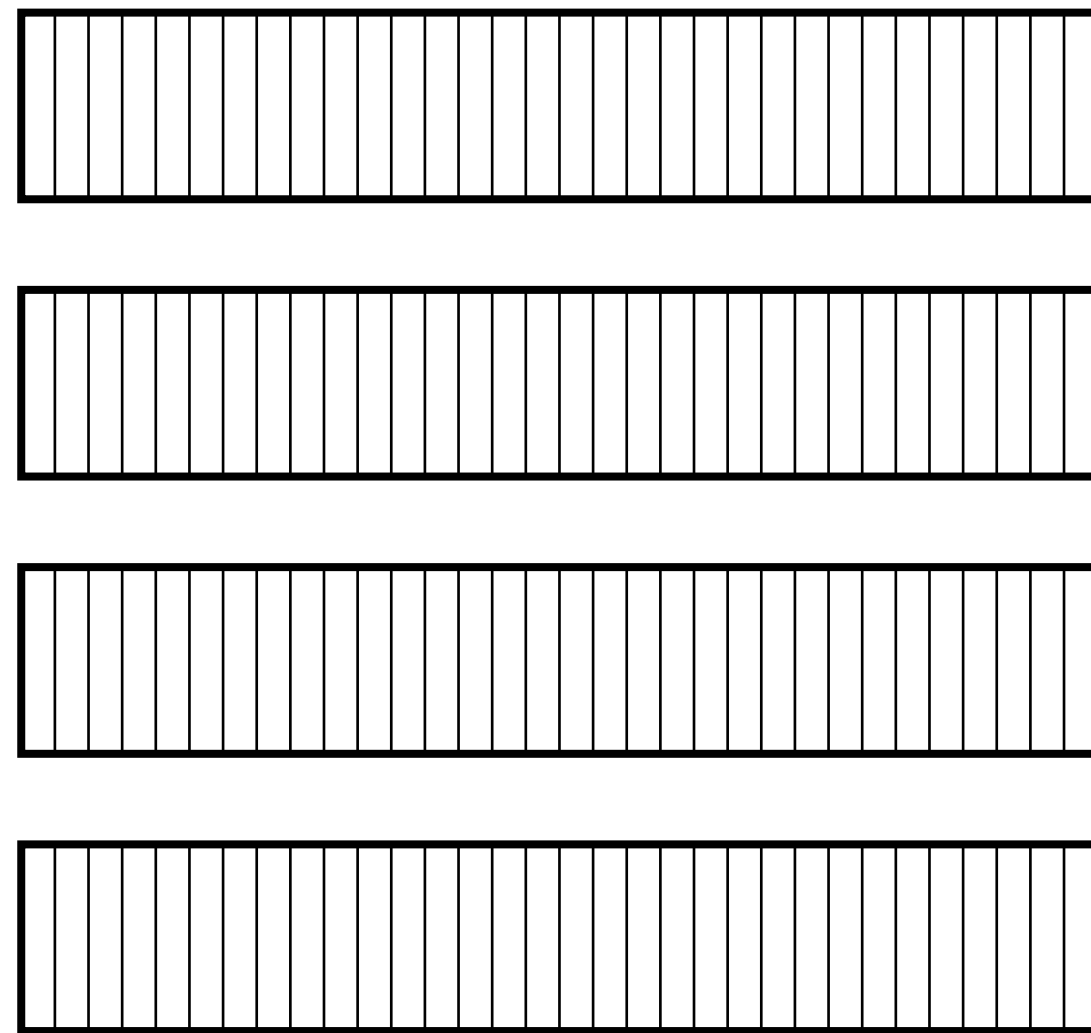


$AC(2i+1)$



Structural observations

We call ***i*-identical** an internal state of Shadow in which *i* bundles are equal.



$$S(y^3+c) = S(y^3)+c'$$

$$S(y^2+c) = S(y^2)+c'$$

$$S(y^1+c) = S(y^1)+c'$$

$$S(y^0+c) = S(y^0)+c'$$

Structural observations

We call ***i*-identical** an internal state of Shadow in which *i* bundles are equal.

probabilities of an *i*-identical state at step *s*

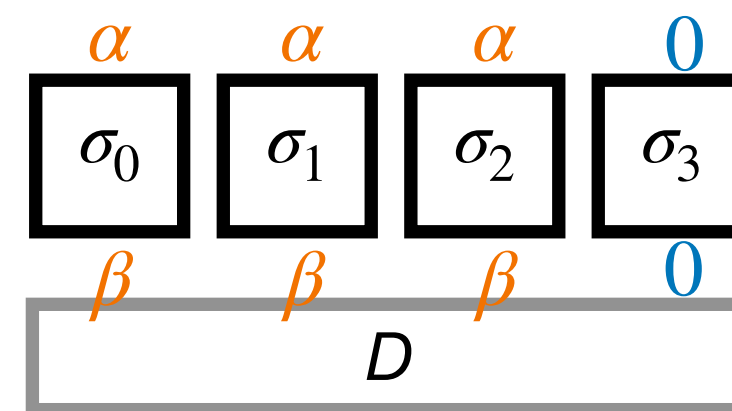
<i>s</i>	0	1	2	3	4
<i>i</i> =4	0	0	2^{-12}	2^{-8}	0
<i>i</i> =3	0	0	2^{-9}	2^{-6}	0
<i>i</i> =2	0	0	2^{-6}	2^{-4}	0

Distinguisher

Distinguisher on 6 steps of Shadow-512

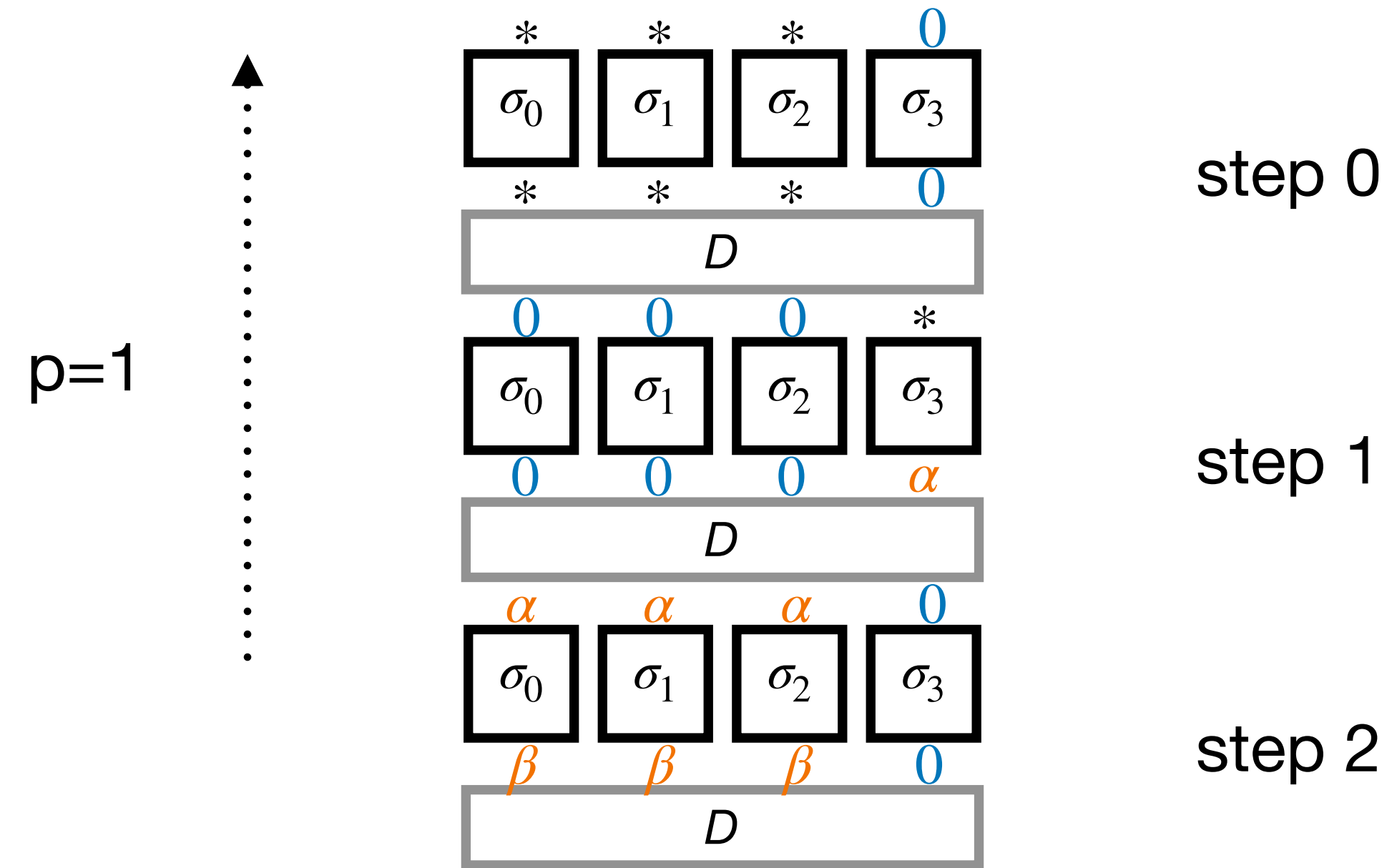
◦ $x \oplus x' = (*, *, *, 0)$ and $\text{shadow}(x) \oplus \text{shadow}(x') = D(0, 0, 0, *)$

◦ Generic cost 2^{-64} vs $2^{-16.245}$ here

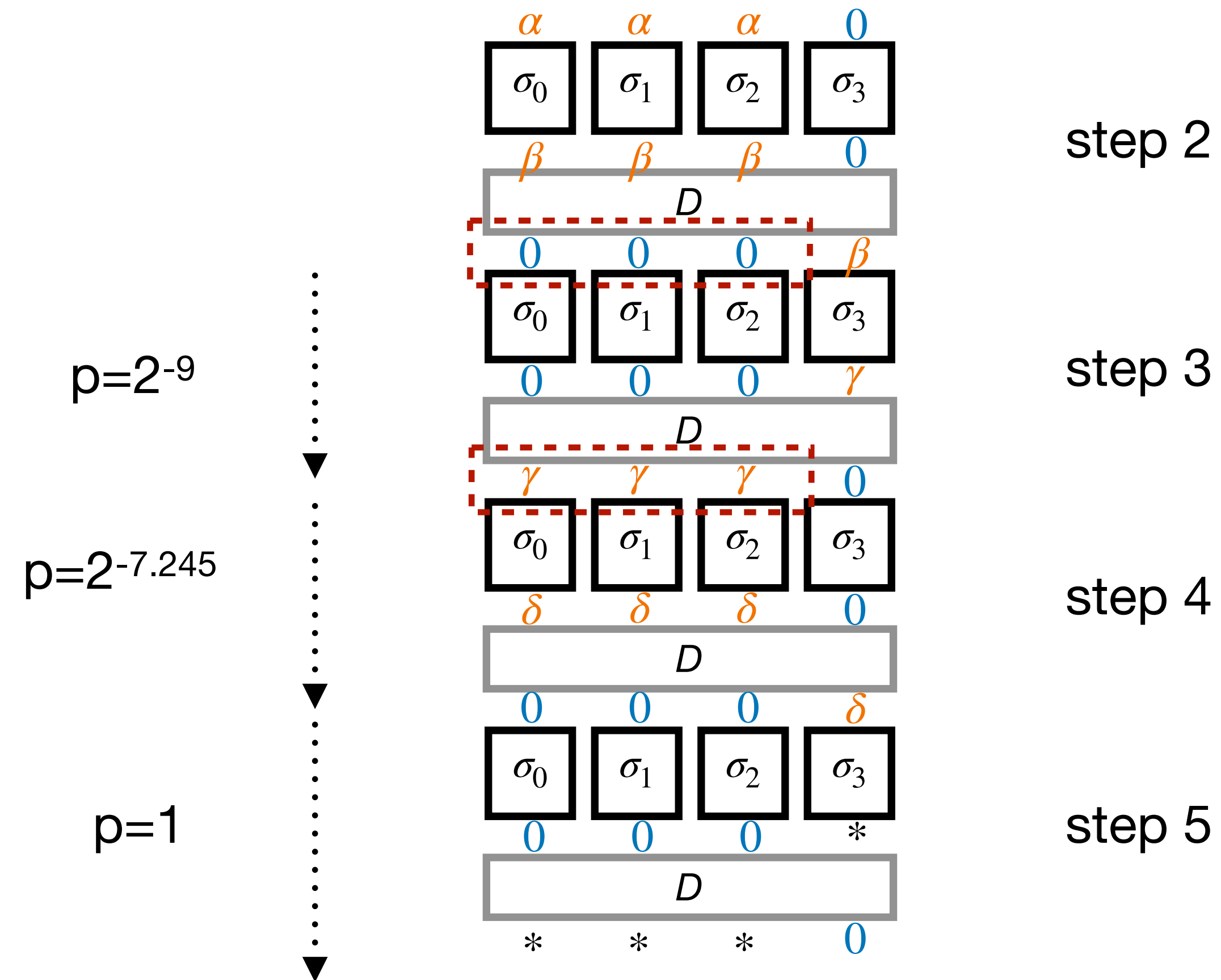


step 2

Distinguisher on 6 steps of Shadow-512



Distinguisher on 6 steps of Shadow-512



Some details

- Constructing a pair for **step 2**:

- $$\sigma_0(x) + \sigma_0(x + \alpha) = \beta$$

$$\sigma_1(x + \epsilon) + \sigma_1(x + \epsilon + \alpha) = \beta$$

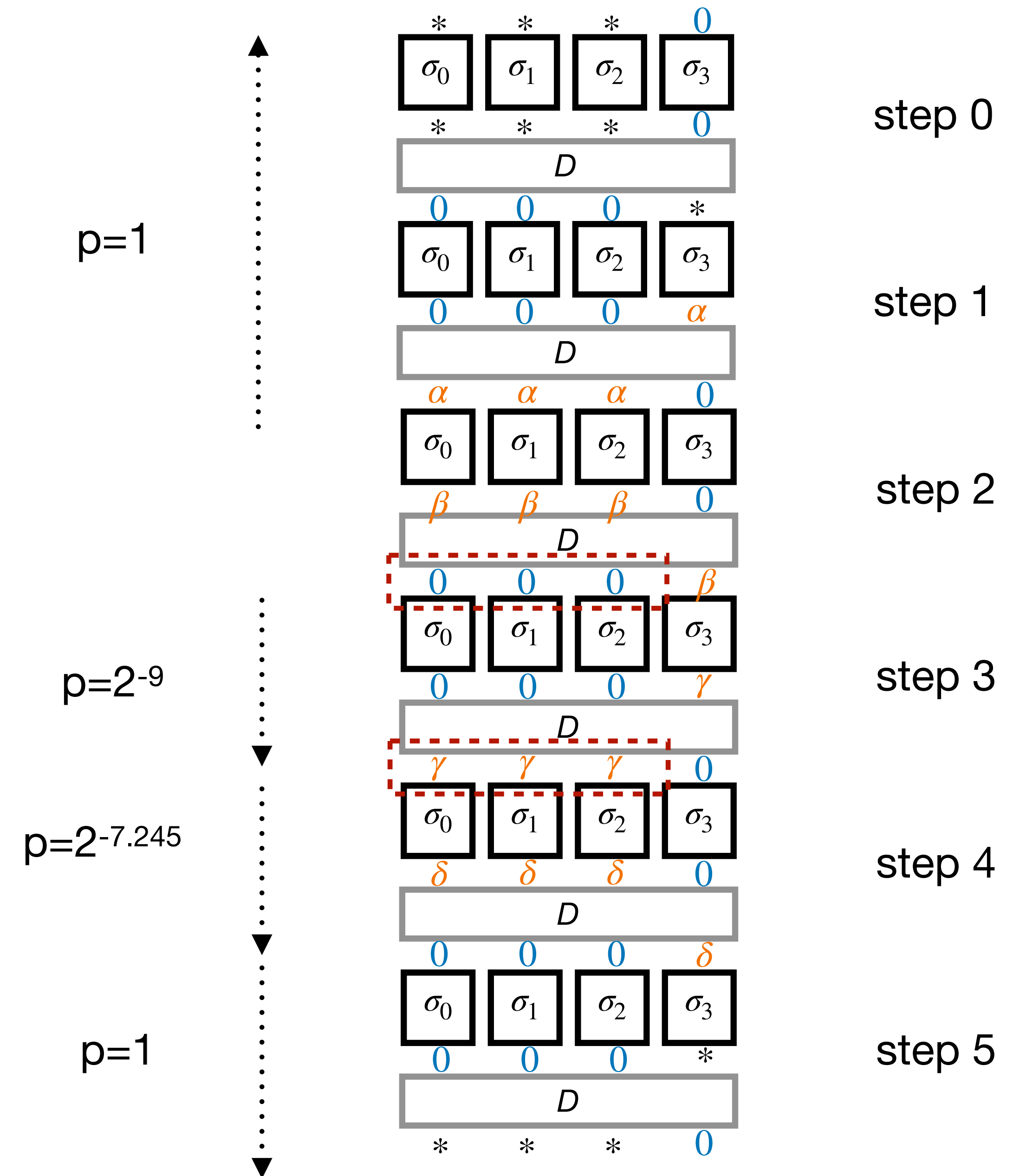
$$\sigma_2(x + \epsilon') + \sigma_2(x + \epsilon' + \alpha) = \beta$$

and **3-identical state at the end of step 2**

- Impact of the constant additions limited to the S-boxes with indices in $\{0,1,2,3\}$
- Bits with indices **22** and **23** in each of the 4 input words of a Super S-box have **no influence** on the output bits with indices in $\{0,1,2,3\}$

$$\nabla = \{a \times e_{22} + b \times e_{23}, a \in \mathbb{F}_2^4, b \in \mathbb{F}_2^4\}$$

For all $\alpha \in \nabla$, all steps and all bundle index i ,
 $\sigma_i(x) + \sigma_i(x + \alpha) = (*, *, \dots, *, 0, 0, 0, 0)$



Some details

- Step 3: probability of a **3-identical state** = 2^{-9}
- Step 4: **difference of the form $(0,0,0,\delta)$ at the end of the step**

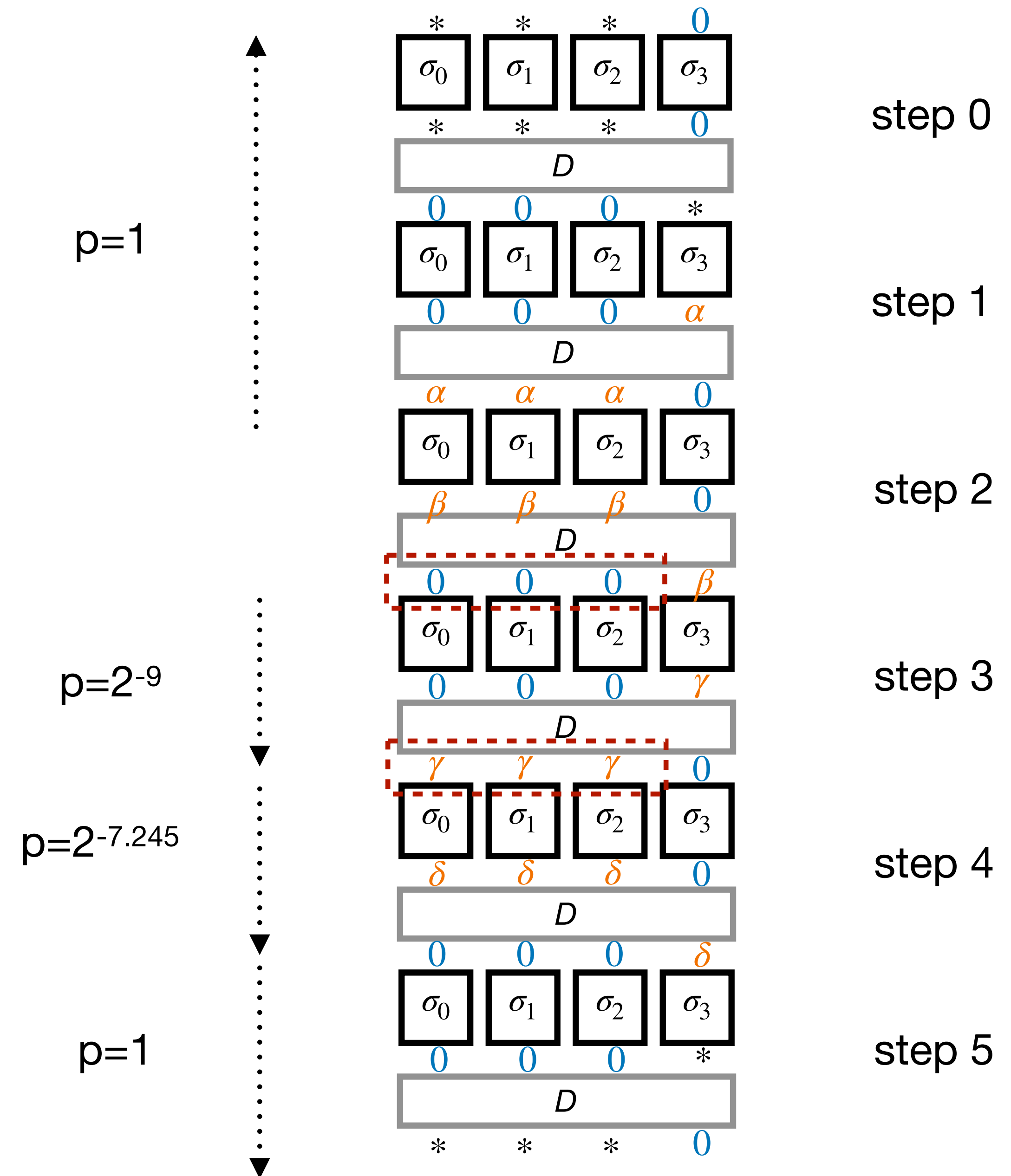
Let (y, y, y, w) and (y', y', y', w) denote two messages after the application of S and L of step 4 then:

$$\begin{aligned}
 S(y^2) \oplus S(y^2 \oplus c) &= S(y^2) \oplus S(y^2 \oplus c) \\
 S(y^1) \oplus S(y^1 \oplus c) &= S(y^1) \oplus S(y^1 \oplus c) \\
 S(y^0) \oplus S(y^0 \oplus c) &= S(y^0) \oplus S(y^0 \oplus c)
 \end{aligned}$$

with $c = 0x5$, probability of $2^{-2.415}$ for each equality

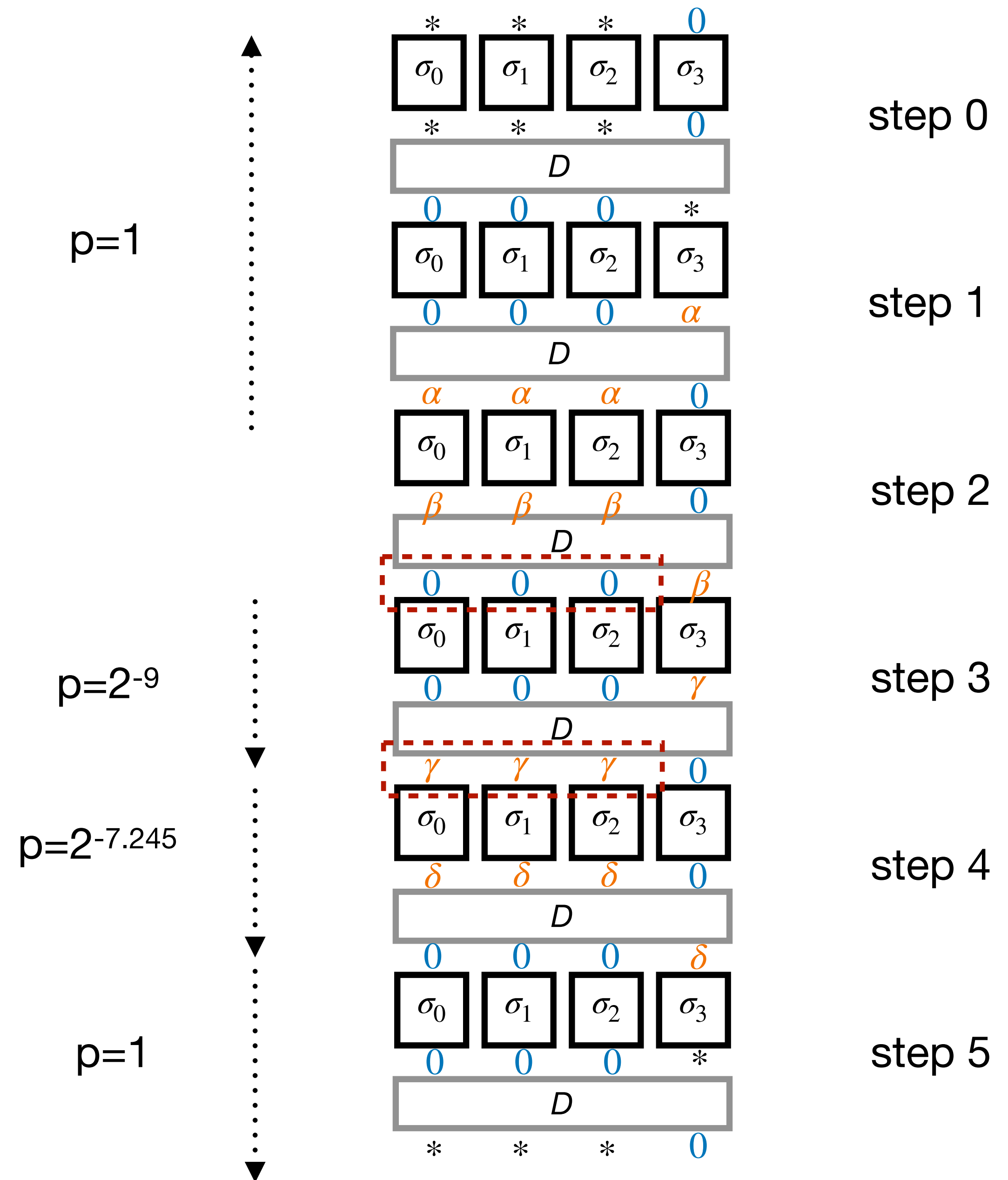
- Step 5 has probability 1

Total probability: $(2^{-2.415})^3 \times 2^{-9} = 2^{-16.245}$



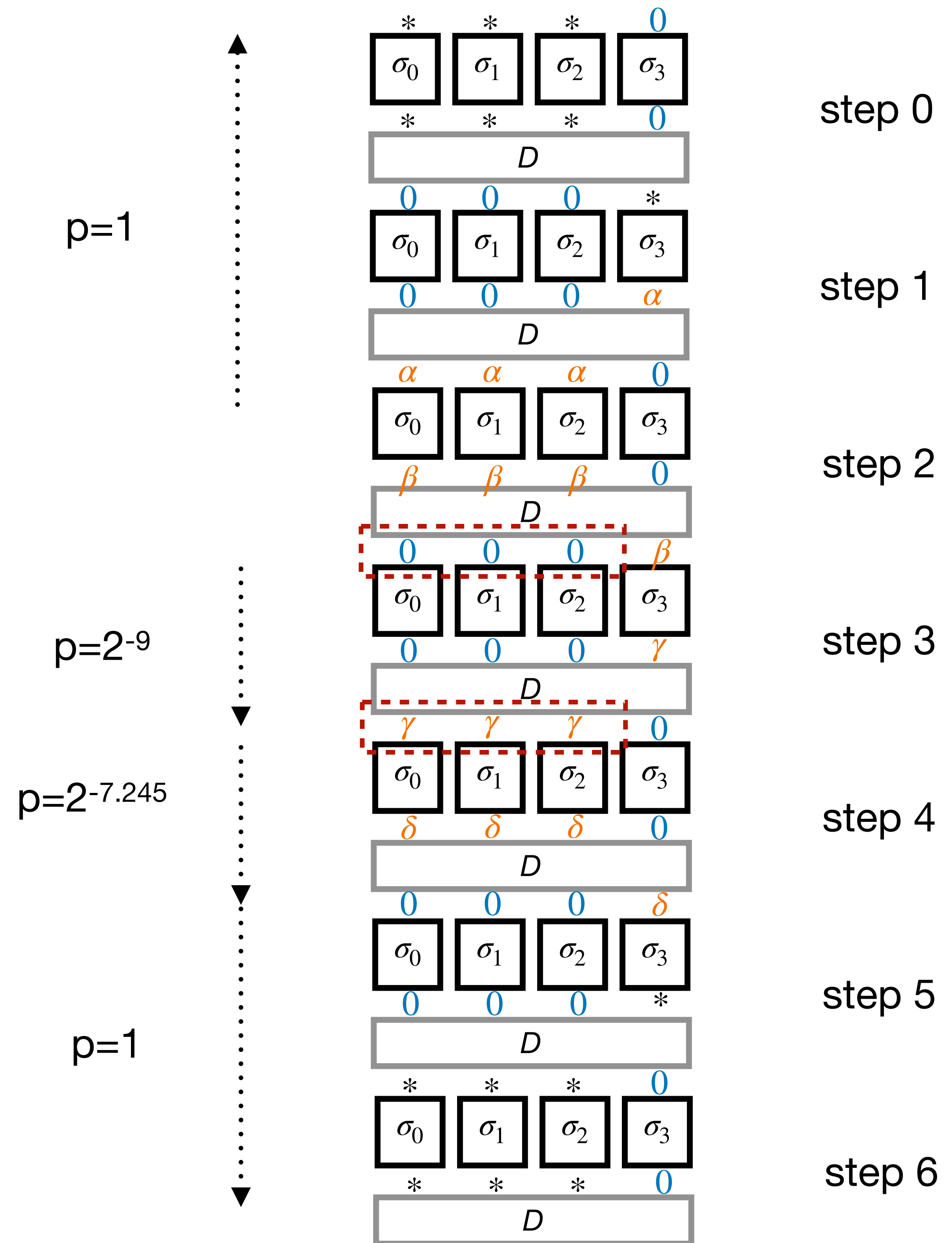
Summary

1. Select a difference $\alpha \in \nabla$.
2. Select a state (y_2, y_2, y_2, z_2) that will be a state after step 2.
3. Invert step 2 on (y_2, y_2, y_2, z_2) , obtaining (x_1, y_1, z_1, t_1) .
4. Invert step 1 on (x_1, y_1, z_1, t_1) and $(x_1 \oplus \alpha, y_1 \oplus \alpha, z_1 \oplus \alpha, t_1)$, obtaining (x_0, y_0, z_0, t_0) and (x_0, y_0, z_0, t'_0) .
5. Invert step 0, obtaining a pair of Shadow-512 states with a zero-difference in the last bundle.
6. Return this pair of state. With high probability $\geq 2^{-16.245}$, it satisfies the truncated trail.



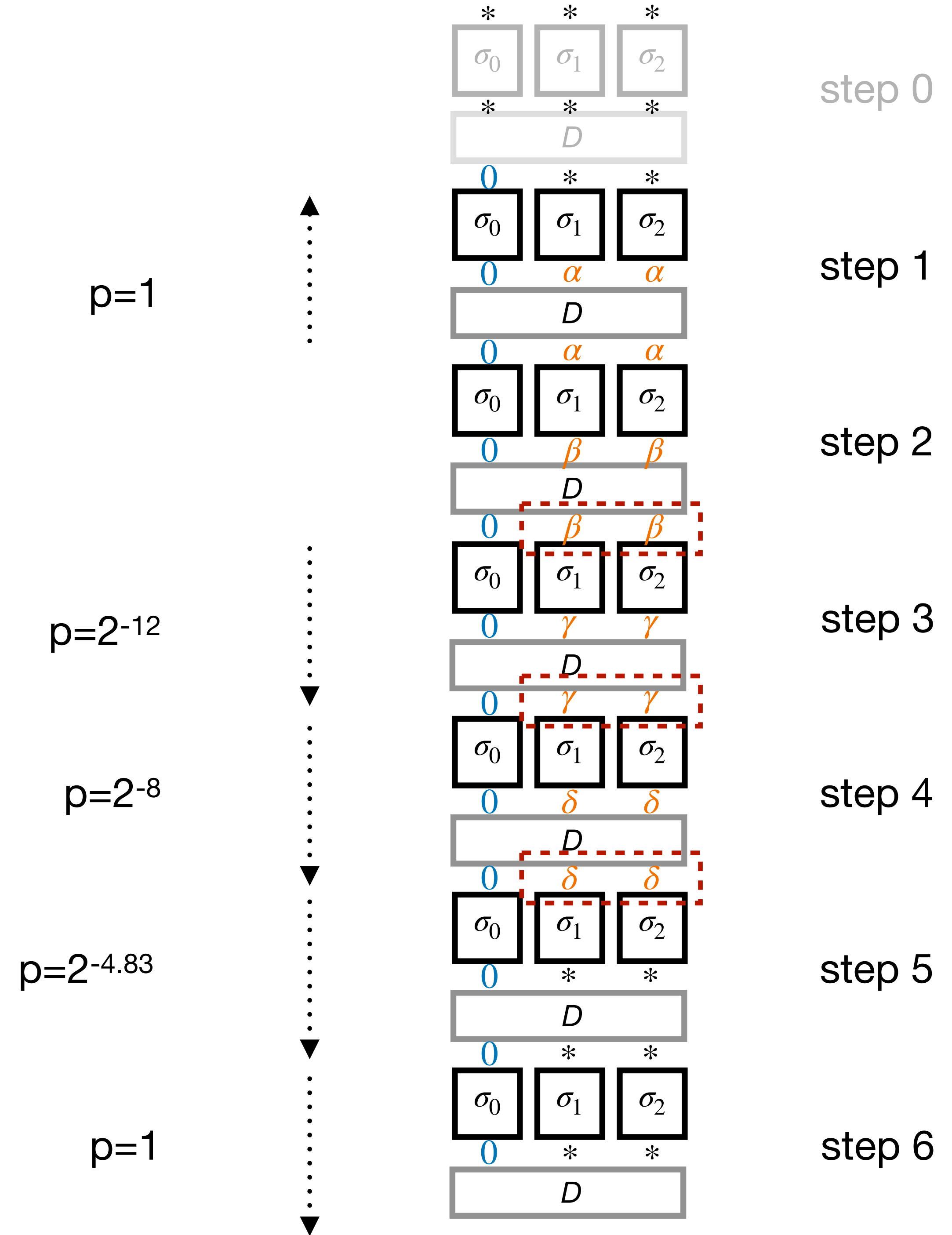
Extension to 7 steps

No extra cost.



The Shadow-384 case

$$D(a, b, c) = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \times \begin{pmatrix} a \\ b \\ c \end{pmatrix}$$



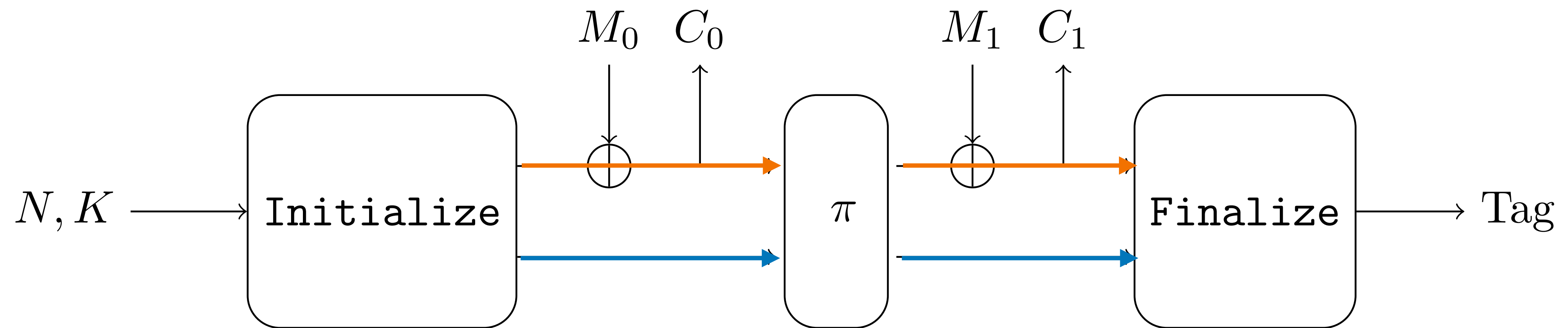
Forgery

Forgery

- “Aggressive parameters”: 8 rounds for Shadow-512
- Shifted version (step 2 to step 5)
- Same nonce used 3 times (nonce misuse scenario) to build collisions: **2 different plaintexts** that yield the **same tag**

Forgery

S1P mode in our attack setting

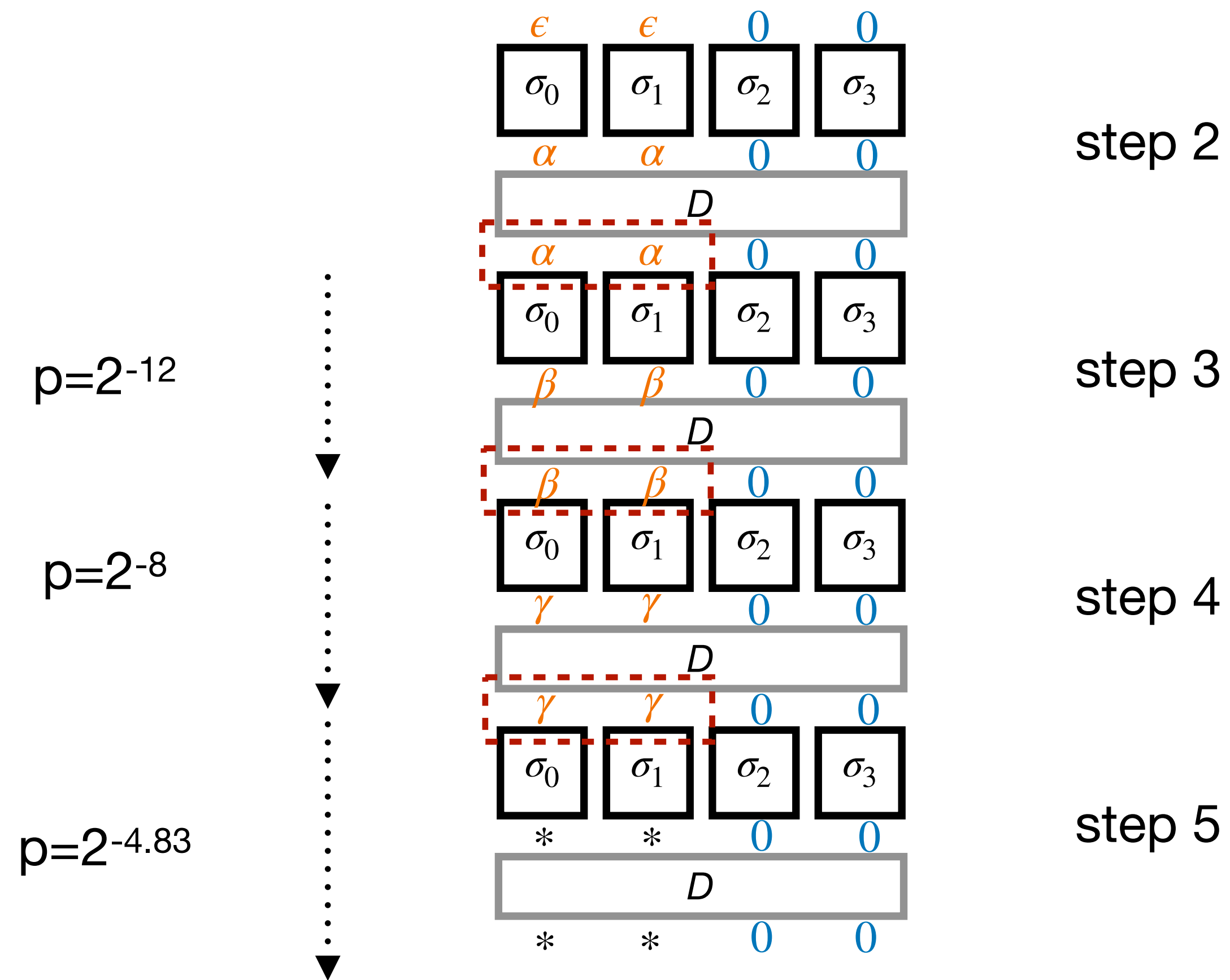


rate: bundle 0, 1

capacity: bundle 2, 3, **not visible**

Forgery

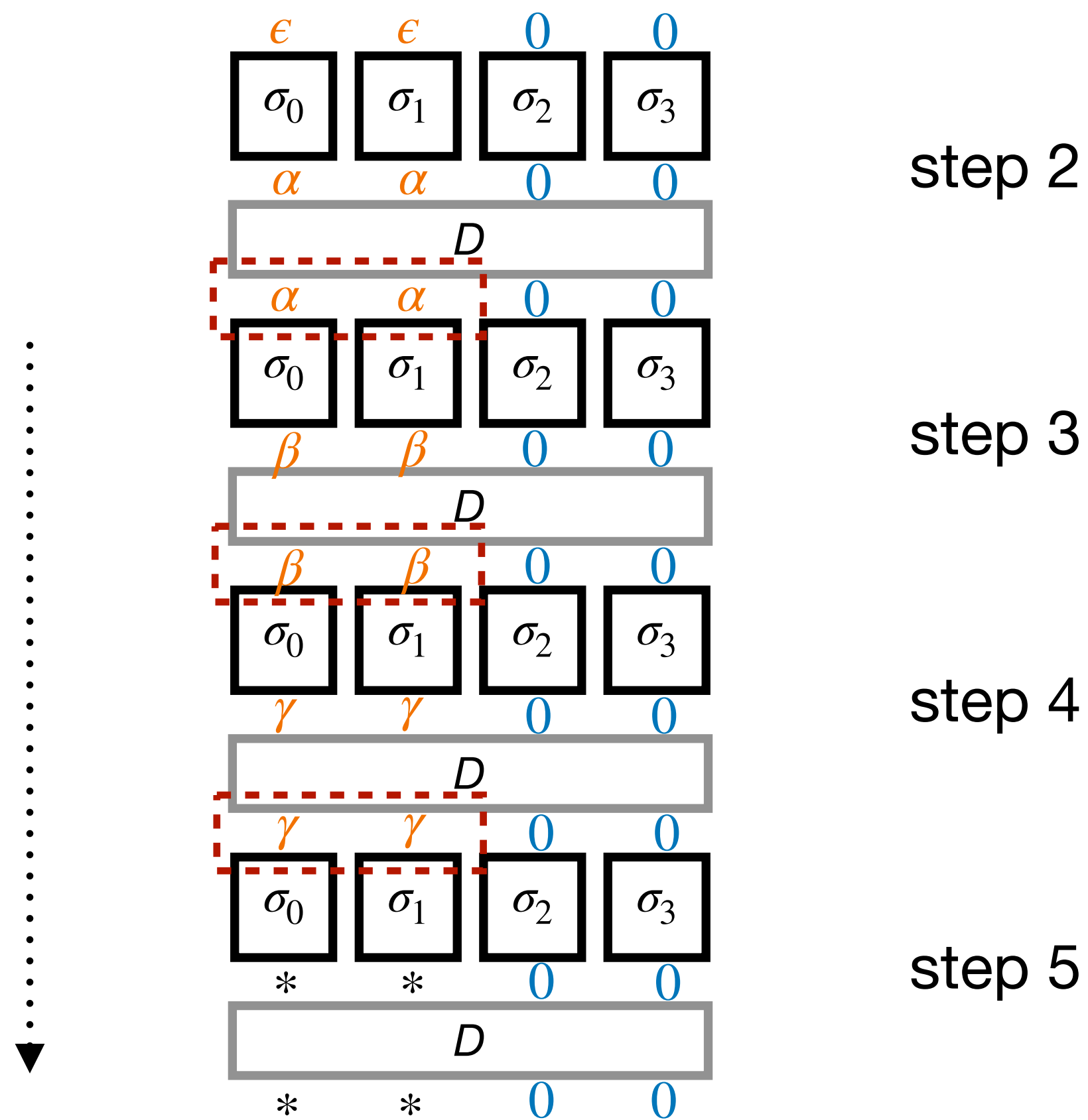
Differential trail



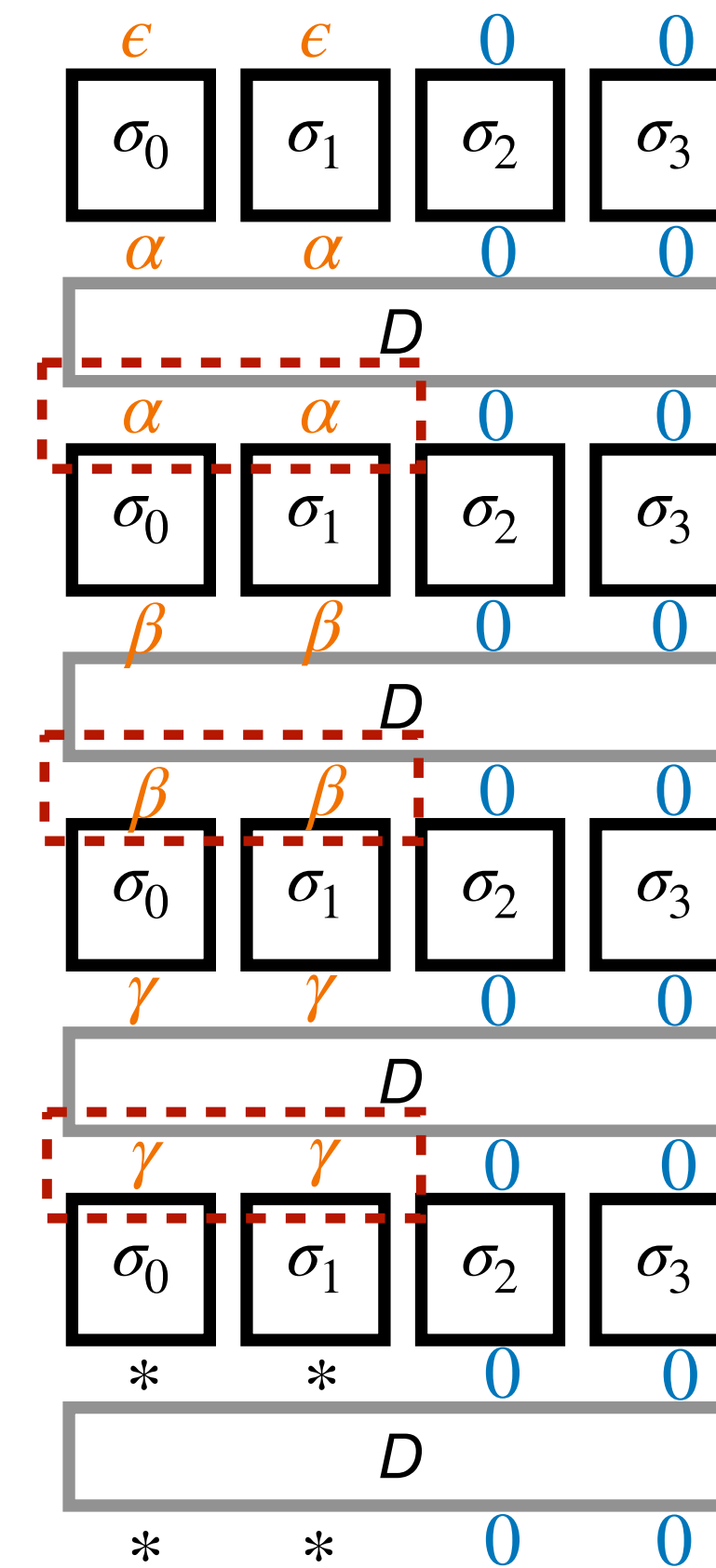
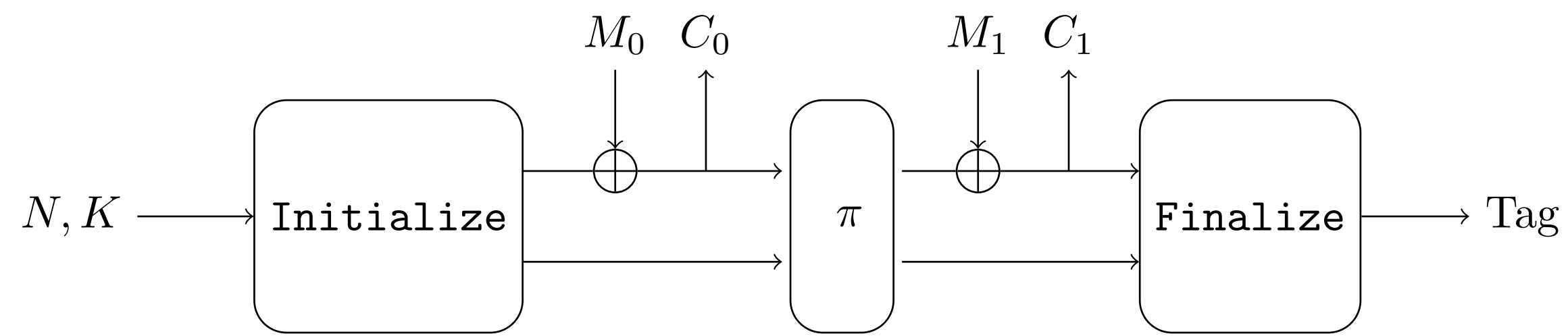
Forgery

Differential trail

Total probability: $2^{-24.83}$



Forgery Outline



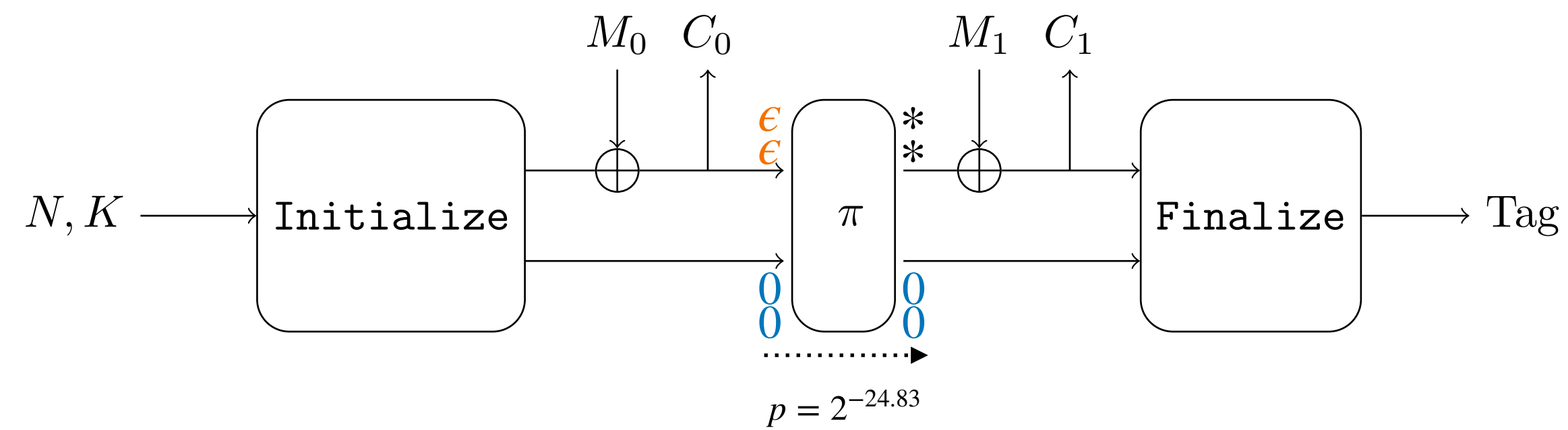
step 2

step 3

step 4

step 5

Forgery Attack Outline

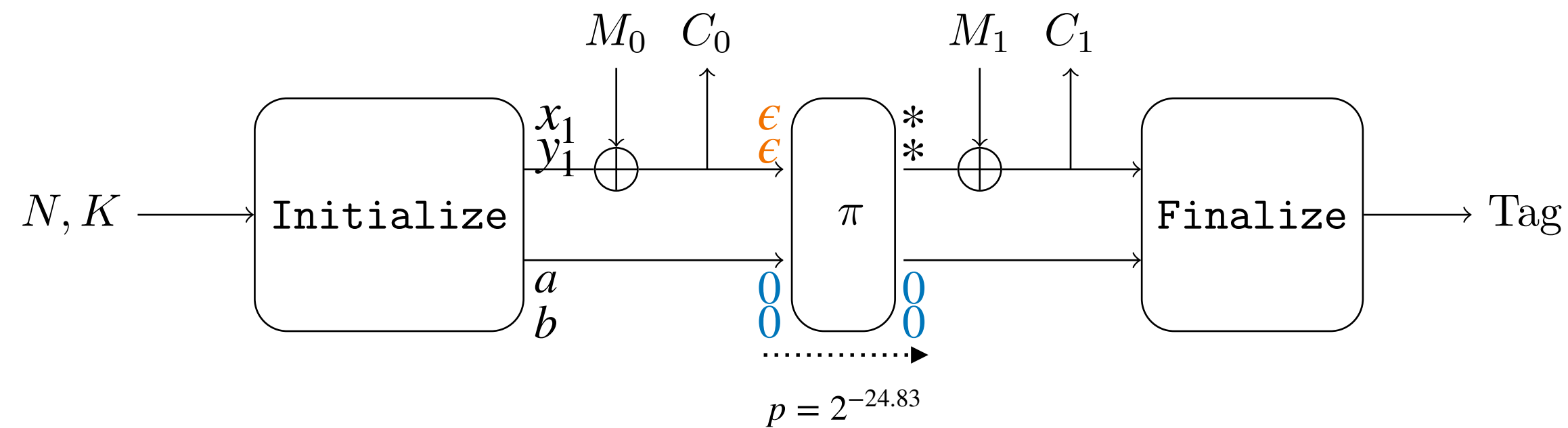


2 different plaintexts that yield the **same tag**



(M_0, M_1) and **(M'_0, M'_1)** that yield a
 $(0,0,0,0)$ difference after π

Forgery Attack Outline



1. **Query 1**: encrypt a two-block (4 bundles) message $(0,0)(0,0)$ to recover the 2-bundle rate value after **Initialize** (x_1, y_1) (\mathbf{C}_0).
2. Generate two pairs of **rate bundles** $(x'_1, y'_1), (x''_1, y''_1)$ that satisfy the truncated trail with probability p .
3. **Query 2 and 3**: get the difference after π .
 - Encrypt $(x_1 \oplus x'_1, y_1 \oplus y'_1), (0,0)$ to obtain the **value of the rate after π on (x'_1, y'_1, a, b)** , denoted by (c'_2, c'_3) (\mathbf{C}_1).
 - Encrypt $(x_1 \oplus x''_1, y_1 \oplus y''_1), (0,0)$ to obtain the **value of the rate after π on (x''_1, y''_1, a, b)** , denoted by (c''_2, c''_3) (\mathbf{C}_1).
4. Cancel out the difference after π .
 - $(x_1 \oplus x'_1, y_1 \oplus y'_1), (c'_2, c'_3)$ and $(x_1 \oplus x''_1, y_1 \oplus y''_1), (c''_2, c''_3)$ yield the same internal state before **Finalize** with probability $p \simeq 2^{-24.83}$.

Conclusion

- Summary of our work:
 - **Practical distinguishers** of the full 6-step version of [Shadow-512](#) and [Shadow-384](#) (shifted)
 - **Practical forgeries** with 4-step Shadow for the S1P mode of operation (nonce misuse scenario)
- After our results, the authors proposed **Spook v2** [ToSC special Issue] :
 - D matrix replaced with an efficient MDS matrix
 - modification of the round constants of Shadow for more efficiency
 - 2nd mathematical challenge ongoing: <https://www.spook.dev/challenges>
- New criterion for choosing round constants: prevent more than invariant subspaces attacks

Thank you!