Improved and Multiple Linear Cryptanalysis of Reduced Round Serpent

Description of the Linear Approximations

B. Collard, F.-X. Standaert*, J.-J. Quisquater

UCL Crypto Group, Microelectronics Laboratory, Louvain-la-Neuve, Belgium

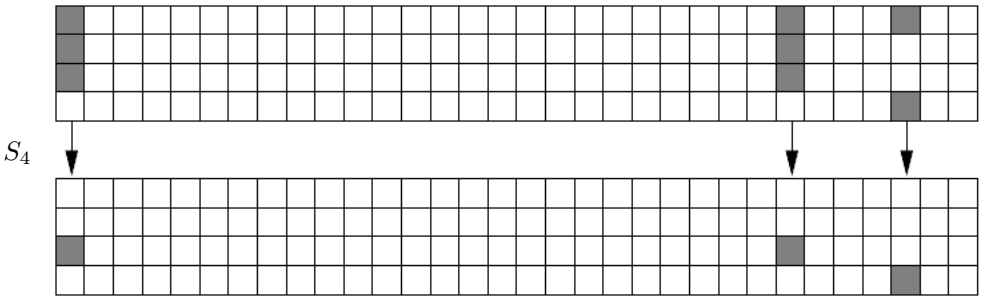
This appendix contains the precise description of the linear approximations used in the attacks scenarios against the reduced round cipher Serpent described in:

- B. Collard, F.-X. Standaert, J.-J. Quisquater, *Improved and Multiple Linear Cryptanalysis of Reduced Round Serpent*, proceedings of InsCrypt 2007.

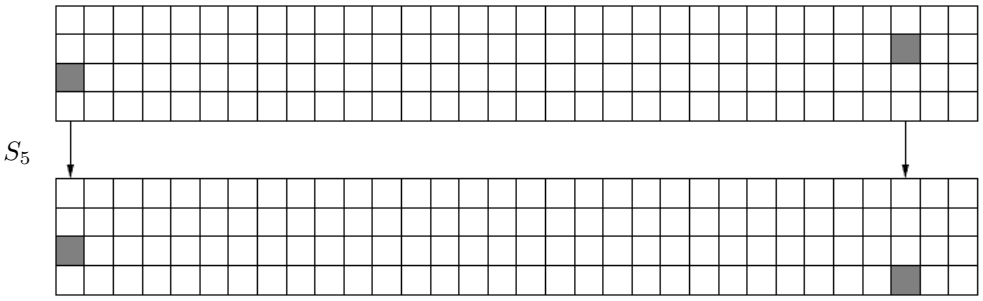
* Postdoctoral researcher of the Belgian fund for scientific research (FNRS).

A 6-round approximation

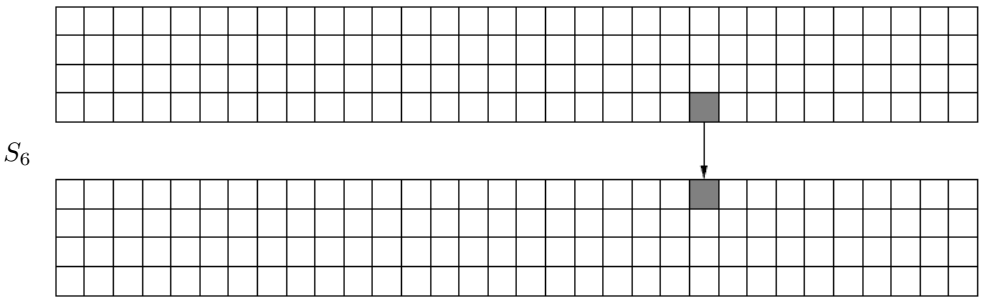
The convention adopted throughout this appendix to describe the linear approximations are also used in [2–4]. We refer to these papers for more details. This section presents a 6-round approximation returned by our algorithm. The approximation has a bias of 2^{-23}. It starts with S-box 4, and the first round approximation holds with a bias of 2^{-4}:



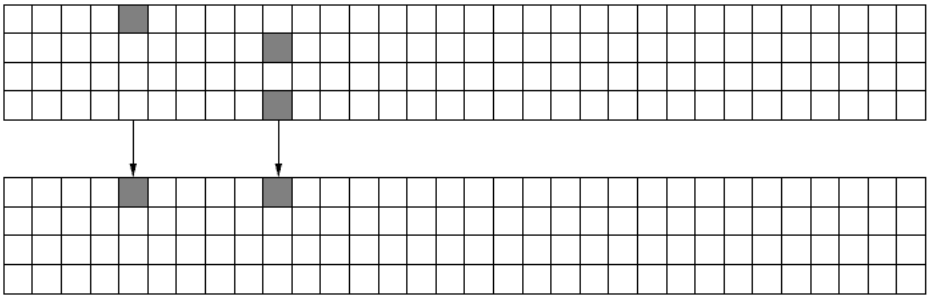
After the linear transformation and the application of S_5, we get the following approximation with bias 2^{-4}:



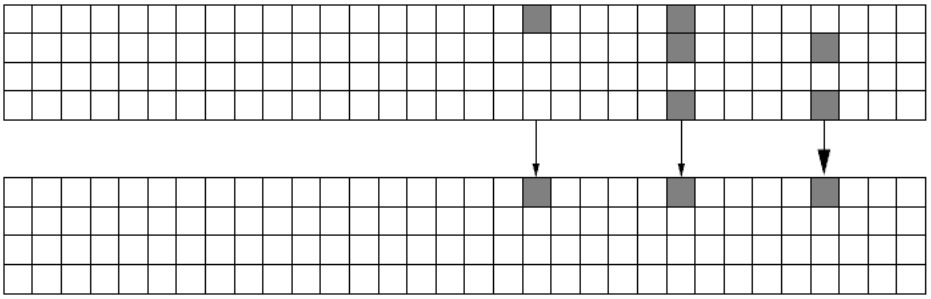
After the linear transformation and the application of S_6, we get the following approximation with bias 2^{-3}:



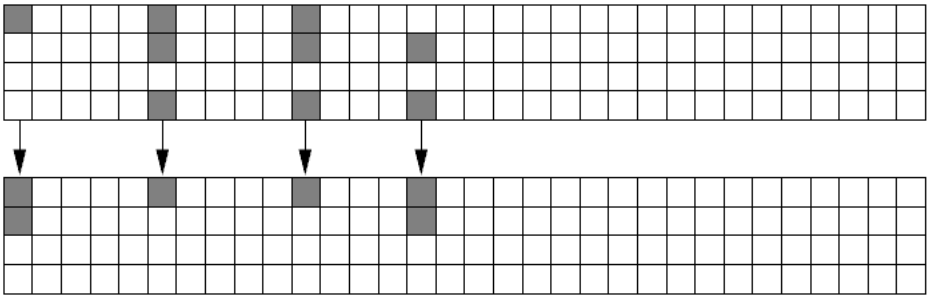
After the linear transformation and the application of S_7, we get the following approximation with bias 2^{-5}:



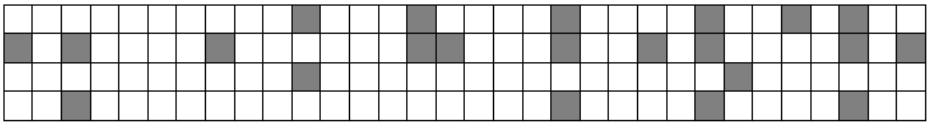
After the linear transformation and the application of S_0, we get the following approximation with bias 2^{-6}:



After the linear transformation and the application of S_1, we get the following approximation with bias 2^{-5}:

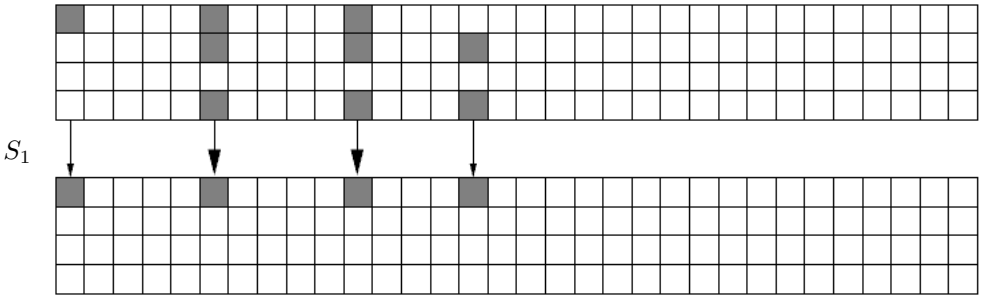


After the linear transformation, we get the following output mask with 13 active S-boxes:

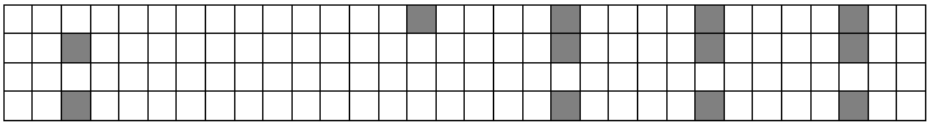


The total bias of the approximation is thus $2^5 \cdot 2^{-4} \cdot 2^{-5} \cdot 2^{-3} \cdot 2^{-5} \cdot 2^{-6} \cdot 2^{-5} = 2^{-23}$. In the first round, any input of the 3 active S-boxes can be replaced by another mask, provided that the biases are left unchanged. It appears that there are 2 such masks for each active S-box. In the last round, there are 4 active S-boxes. Among them, two S-boxes have 4 output masks with maximal bias for

the given input, the two other S-boxes have 2 output masks with maximal bias for the given input. We can consequently generate $2^3 \cdot 4^2 \cdot 2^2 = 2^9$ approximations with the same bias. Alternatively, we can decrease the number of active S-boxes by replacing the last round approximation by the following one:



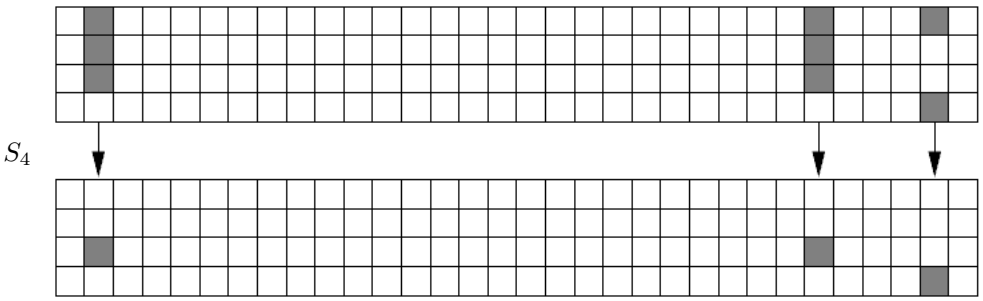
After the linear transformation, we get the following output mask with only 5 active S-boxes:



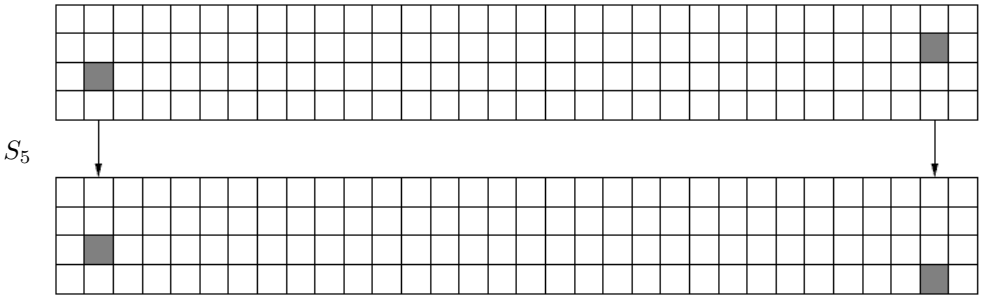
However, the bias of the modified approximation equals 2^{-25}.

B 7-round approximation

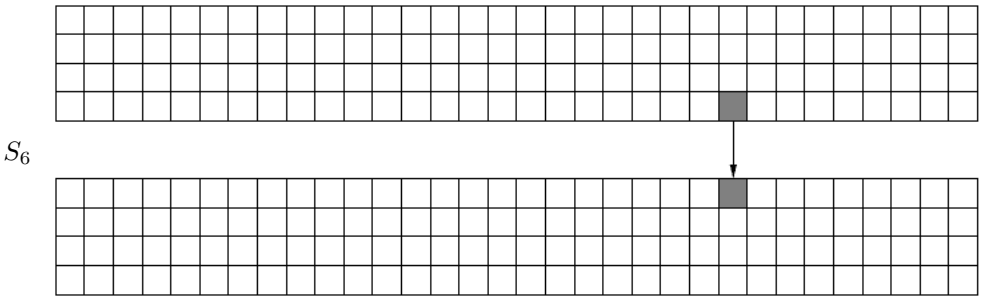
The approximation has a bias of 2^{-30}. It starts with S-box 4, and the first round approximation holds with a bias of 2^{-4}:



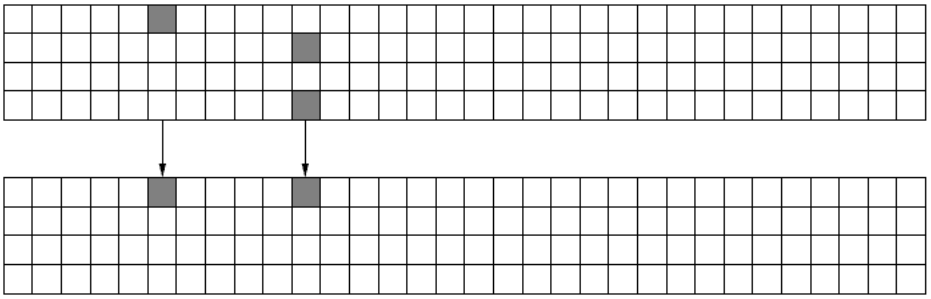
After the linear transformation and the application of S_5, we get the following approximation with bias 2^{-5}:



After the linear transformation and the application of S_6, we get the following approximation with bias 2^{-3}:

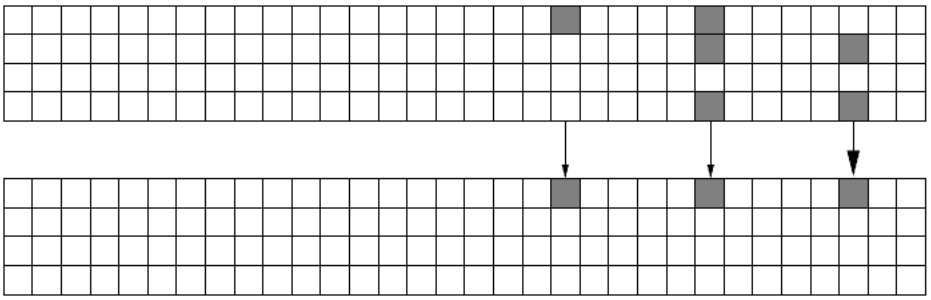


After the linear transformation and the application of S_7, we get the following approximation with bias 2^{-5}:



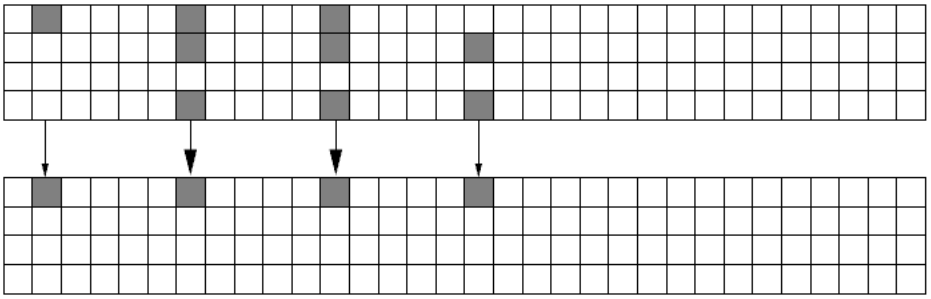
S_7 $Prob = 1/2 - 2^{-5}$

After the linear transformation and the application of S_0, we get the following approximation with bias 2^{-6}:



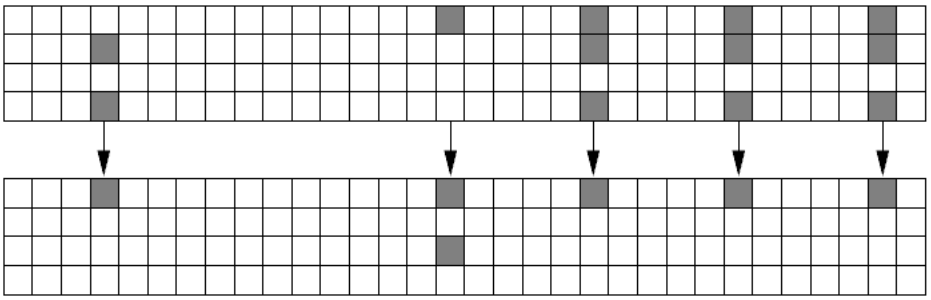
S_0 $Prob = 1/2 + 2^{-6}$

After the linear transformation and the application of S_1, we get the following approximation with bias 2^{-7}:



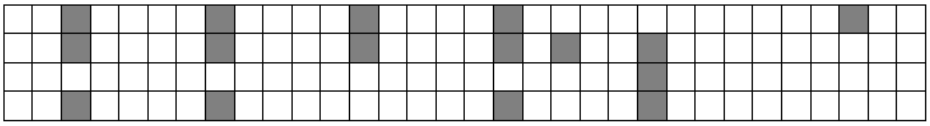
S_1 $Prob = 1/2 - 2^{-7}$

After the linear transformation and the application of S_2, we get the following approximation with bias 2^{-6}:



S_2 $Prob = 1/2 - 2^{-6}$

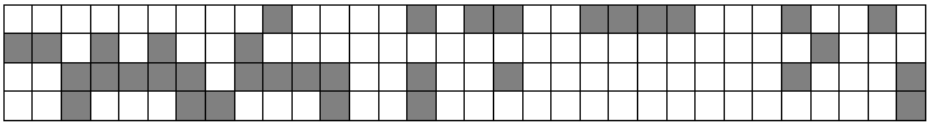
After the last linear transformation, we get the following output mask with only 5 active S-boxes:



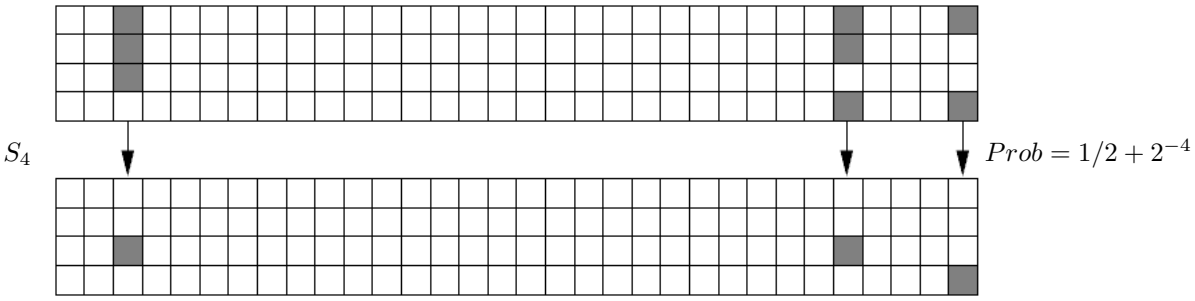
The total bias of the approximation is thus $2^6 \cdot 2^{-4} \cdot 2^{-5} \cdot 2^{-3} \cdot 2^{-5} \cdot 2^{-6} \cdot 2^{-7} \cdot 2^{-6} = 2^{-30}$. In the first round, any input of the 3 active S-boxes can be replaced by another mask, provided that the biases are left unchanged. There are 2 such masks for each active S-box. In the last round, there are 5 active S-boxes. Among them, one has 4 output masks with maximal bias for the given input, the others have 2 output masks with maximal bias for the given input. We can consequently generate $2^3 \cdot 4 \cdot 2^4 = 2^9$ approximations with the same bias.

C 8-round approximation

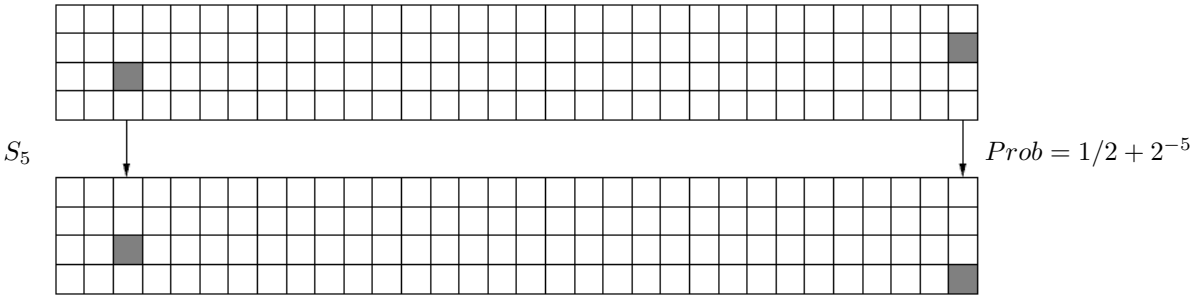
This section describes an 8-round approximation with bias 2^{-37}. In contrast with the other approximations presented in this report, it can be used to attack a reduced version of Serpent by partial encryption of the *first* round with a keyguess (in all other cases, we must perform a partial decryption of the *last* round). This is because the number of active S-boxes before the first round of the approximation is much smaller than it could possibly be after the linear transformation in the last round. The next figure shows the input mask of the approximation as well as the 23 active S-boxes:



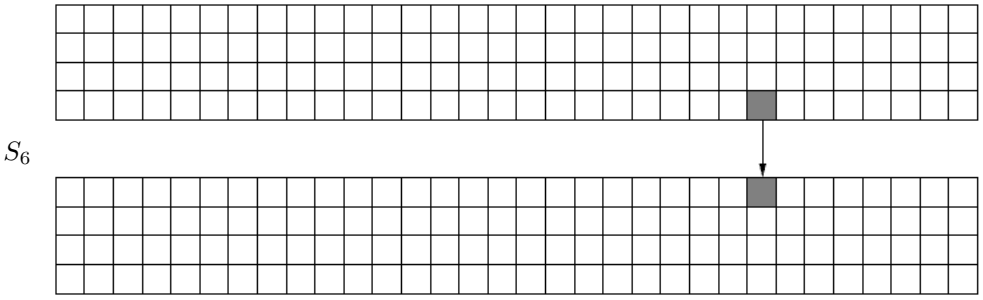
After the linear transformation, the approximation starts with S-box 4, and the first round approximation holds with a bias of 2^{-4}:



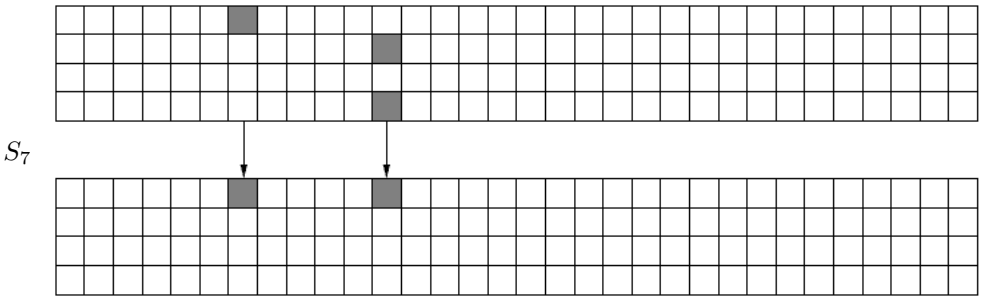
After the linear transformation and the application of S_5, we get the following approximation with bias 2^{-5}:



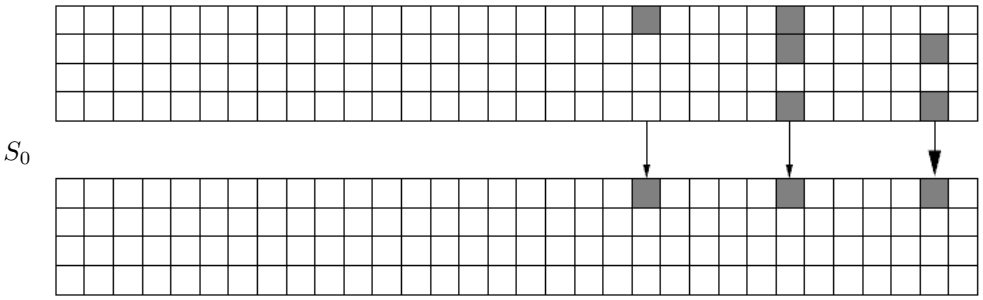
After the linear transformation and the application of S_6, we get the following approximation with bias 2^{-3}:



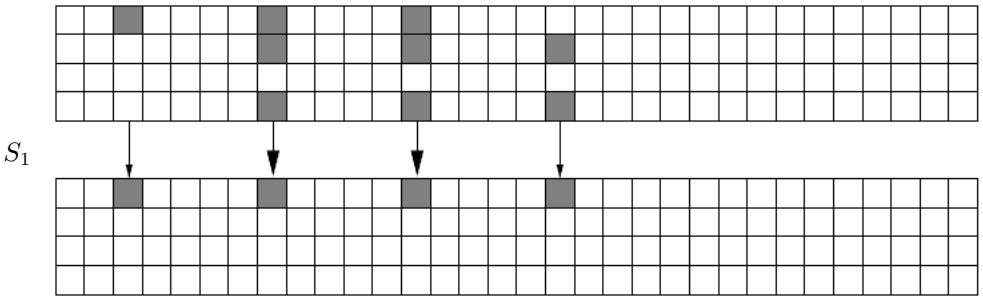
After the linear transformation and the application of S_7, we get the following approximation with bias 2^{-5}:



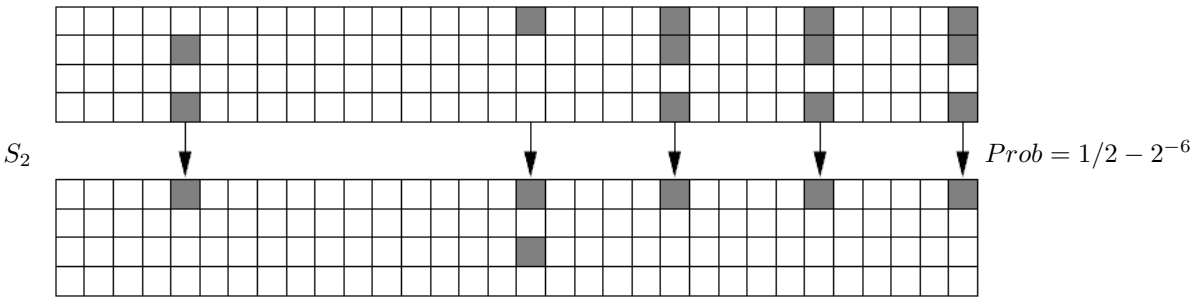
After the linear transformation and the application of S_0, we get the following approximation with bias 2^{-6}:



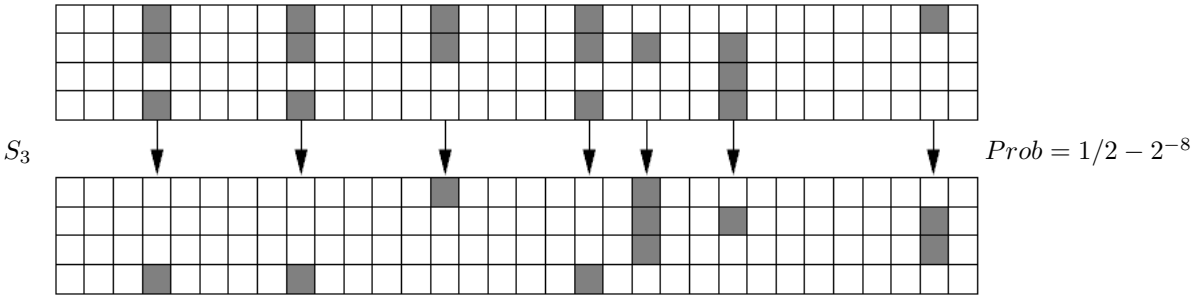
After the linear transformation and the application of S_1, we get the following approximation with bias 2^{-7}:



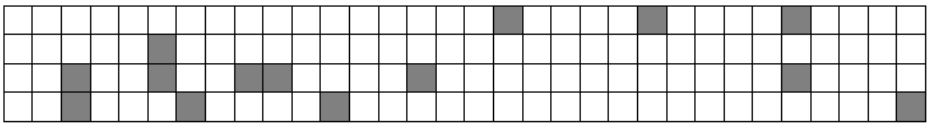
After the linear transformation and the application of S_2, we get the following approximation with bias 2^{-6}:



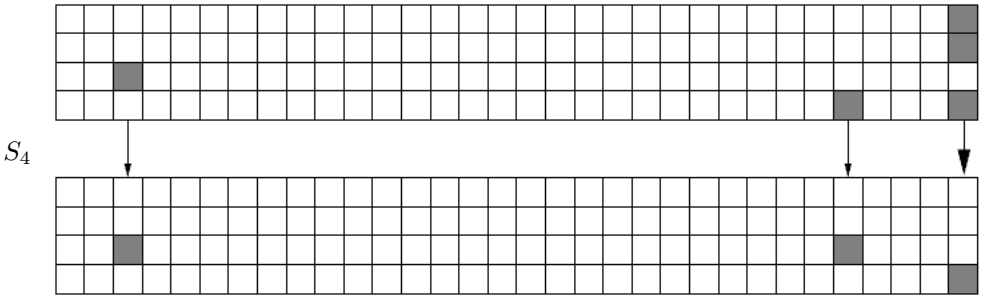
After the linear transformation and the application of S_3, we get the following approximation with bias 2^{-8}:



The total bias of the approximation is thus $2^7 \cdot 2^{-4} \cdot 2^{-5} \cdot 2^{-3} \cdot 2^{-5} \cdot 2^{-6} \cdot 2^{-7} \cdot 2^{-6} \cdot 2^{-8} = 2^{-37}$. In the first round, any input of the 3 active S-boxes can be replaced by another mask, provided that the biases are left unchanged. It appears that there are 2 such masks for each active S-box. In the last round, there are 7 active S-boxes. For these S-boxes, we can choose between two output masks with bias 0.25 for the given input. We can consequently generate $2^3 \cdot 2^7 = 2^{10}$ approximations with the maximal bias. However, the 23 active S-boxes in the input of the linear approximation do not yield effective attacks as the key guess is too large. We can slightly modify the first round approximation to reduce the size of the key guess to $11 \cdot 4$ bits. This gives the following input mask:



After the linear approximation, we use the the following approximation:

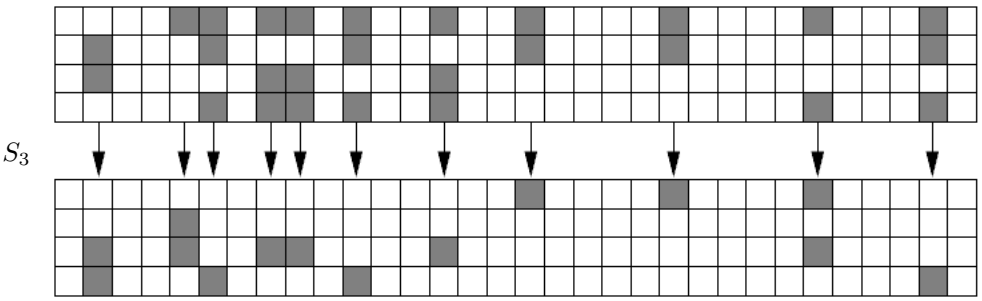


This way, the bias of the approximation is reduced to 2^{-39}.

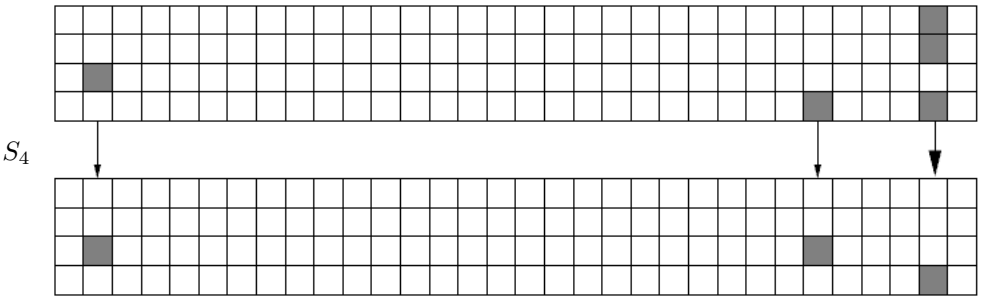
D 9-round approximation

D.1 First approximation

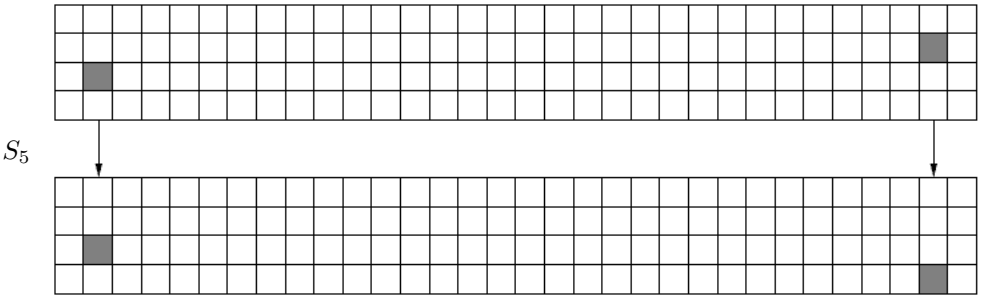
In this section, we present one of the three 9-round approximation with bias 2^{-50} returned by our algorithm. It starts with S-box 3, and the first round approximation holds with a bias of 2^{-12}:



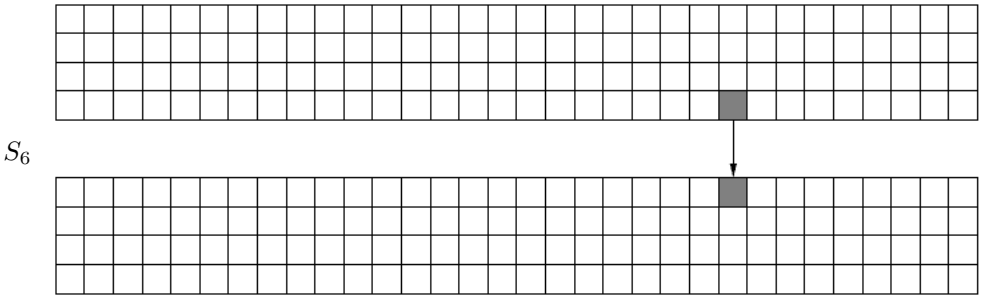
After the linear transformation and the application of S_4, we get the following approximation with bias 2^{-6}:



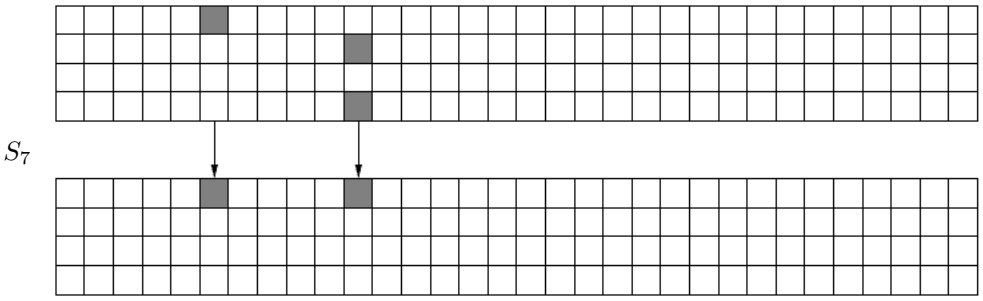
After the linear transformation and the application of S_5, we get the following approximation with bias 2^{-5}:



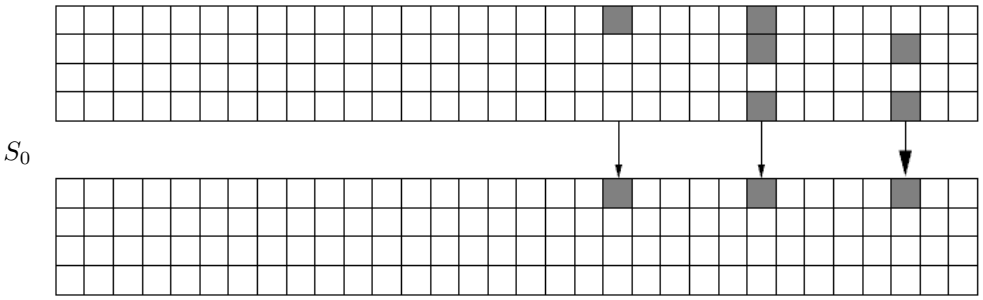
After the linear transformation and the application of S_6, we get the following approximation with bias 2^{-3}:



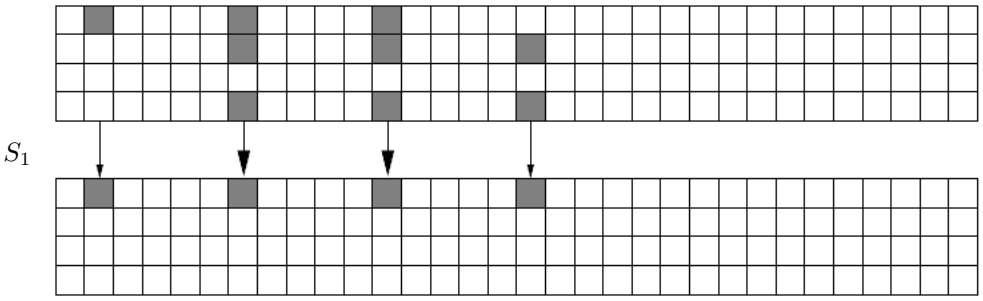
After the linear transformation and the application of S_7, we get the following approximation with bias 2^{-5}:



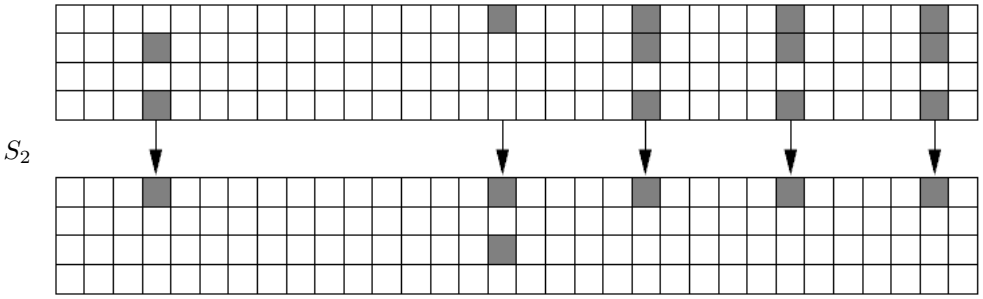
After the linear transformation and the application of S_0, we get the following approximation with bias 2^{-6}:



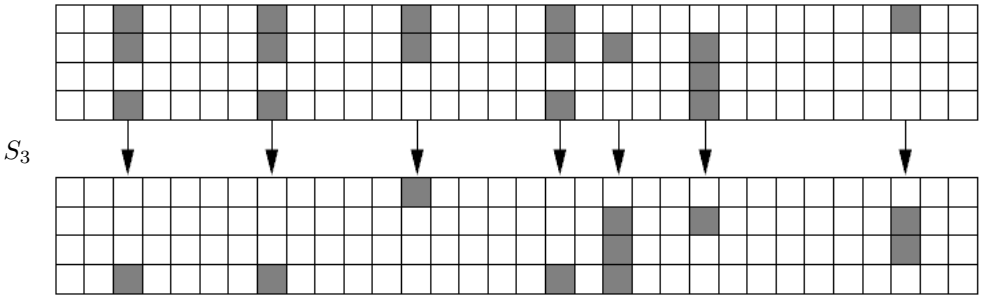
After the linear transformation and the application of S_1, we get the following approximation with bias 2^{-7}:



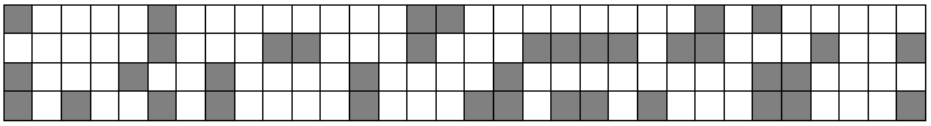
After the linear transformation and the application of S_2, we get the following approximation with bias 2^{-6}:



After the linear transformation and the application of S_3, we get the following approximation with bias 2^{-8}:

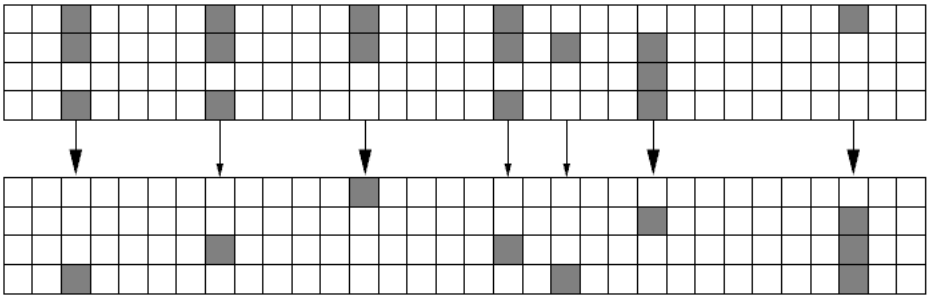


After the last linear transformation, we get the following output mask with 23 active S-boxes:

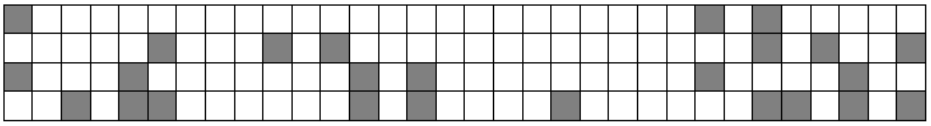


The total bias of the approximation is $2^8 \cdot 2^{-12} \cdot 2^{-6} \cdot 2^{-5} \cdot 2^{-3} \cdot 2^{-5} \cdot 2^{-6} \cdot 2^{-7} \cdot 2^{-6} \cdot 2^{-8} = 2^{-50}$. Any input (resp. output) of the 11 (resp. 7) active S-box in the first round (resp. the last round) can be replaced by another mask, provided that the biases are left unchanged. As there are 2 such masks for each active S-box, We can generate $2^{11} \cdot 2^7 = 2^{18}$ approximations with the bias 2^{50}. This applies to the three approximations with bias 2^{-50} generated by our algorithm, hence the total number rises to $3 \cdot 2^{18}$.

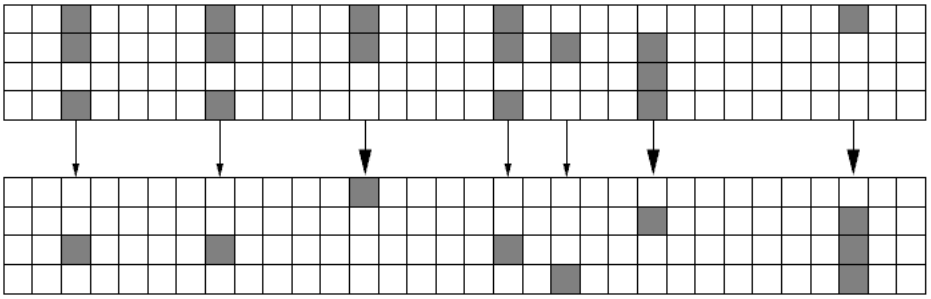
As the number of active S-boxes in the output mask of the approximation is too large, it is possible to slightly modify the last round. Several trade-off can be achieved, whether we wish to reduce the data complexity or the time-memory complexity. Using the following approximation reduces the bias to 2^{-53} but activates only 15 S-boxes:



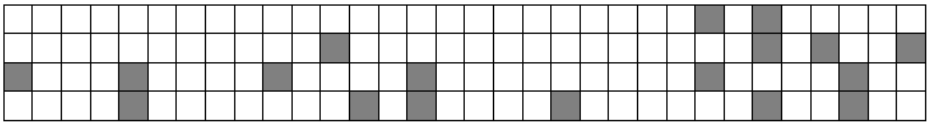
After the last linear transformation, we get the following output mask:



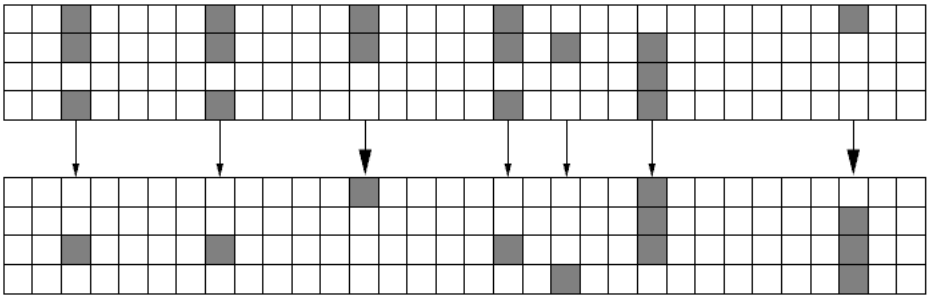
Alternatively, the following approximation has a bias of 2^{-54} but only 12 active S-boxes:



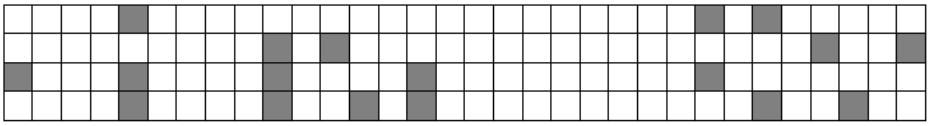
After the last linear transformation, we get the following output mask:



Finally, the following approximation has a bias of 2^{-55} but only 11 active S-boxes:



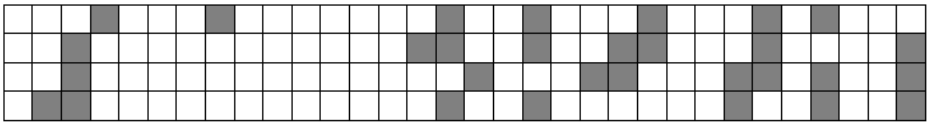
After the last linear transformation, we get the following output mask:



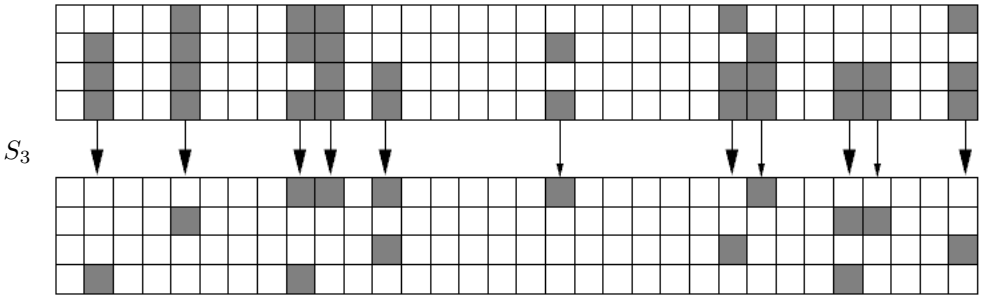
D.2 Second approximation

The following approximation can be used in order to attack 11-round Serpent using a partial decryption of the first and the last rounds. It was designed so as to minimize the number of active S-boxes in its input and output. It has a bias of 2^{-58}.

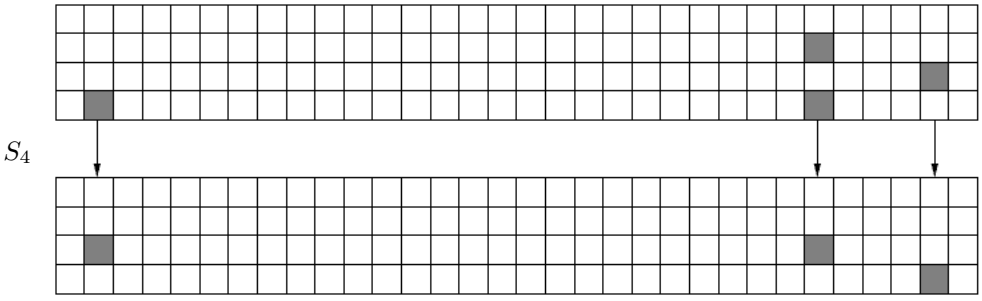
The input mask of the approximation (before the linear transformation) has 15 active S-boxes:



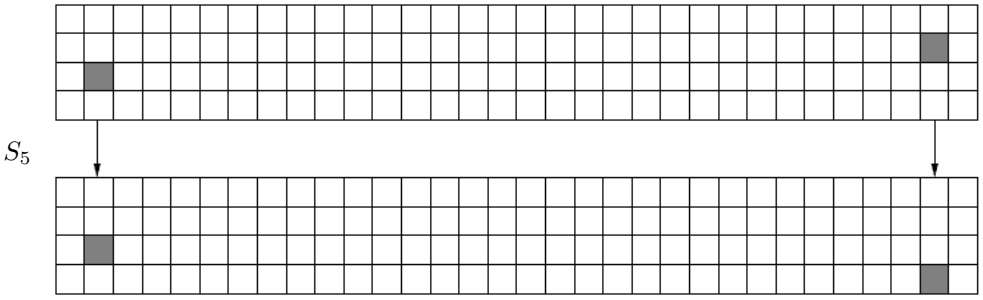
After the linear transformation, the approximation starts with S-box 3, and the first round approximation holds with a bias of 2^{-15}:



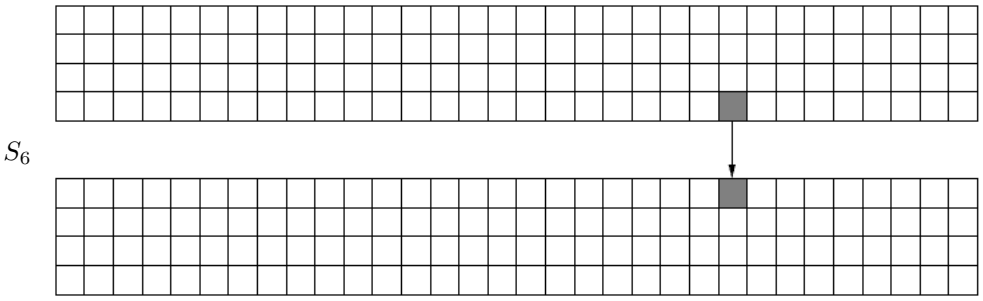
After the linear transformation and the application of S_4, we get the following approximation with bias 2^{-7}:



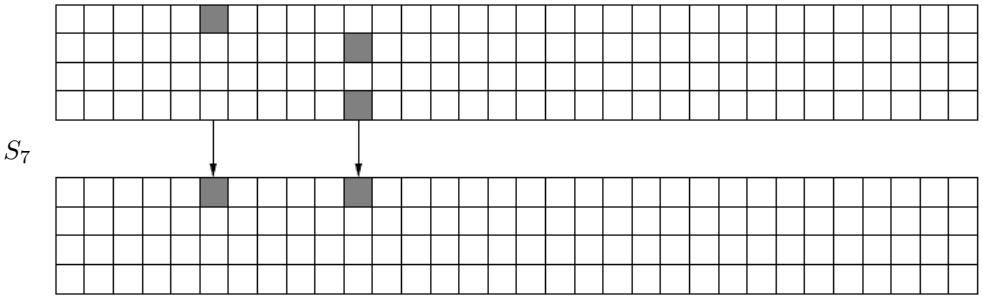
After the linear transformation and the application of S_5, we get the following approximation with bias 2^{-5}:



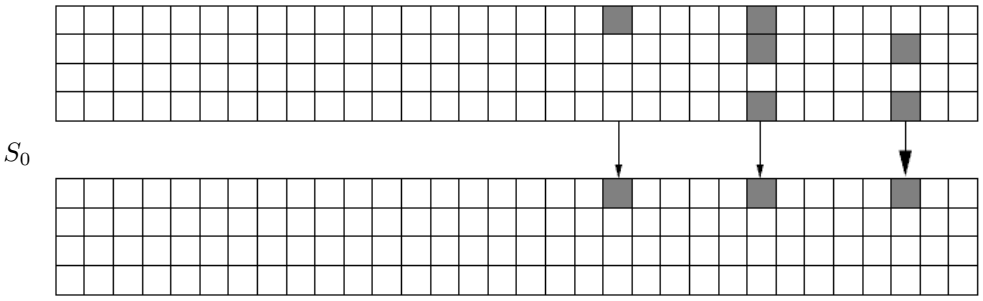
After the linear transformation and the application of S_6, we get the following approximation with bias 2^{-3}:



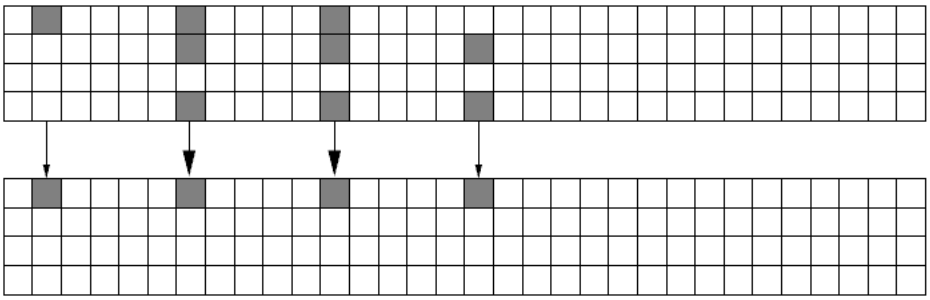
After the linear transformation and the application of S_7, we get the following approximation with bias 2^{-5}:



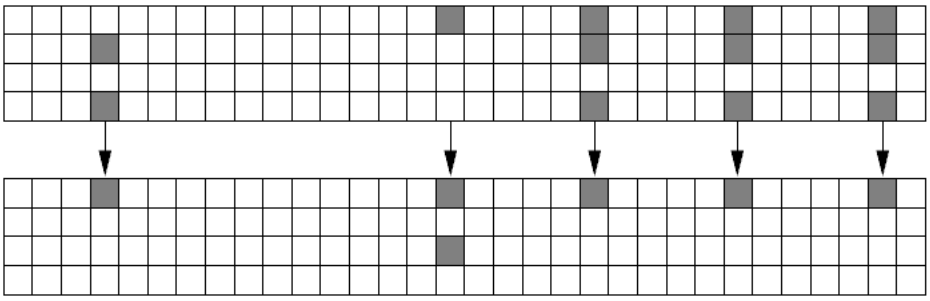
After the linear transformation and the application of S_0, we get the following approximation with bias 2^{-6}:



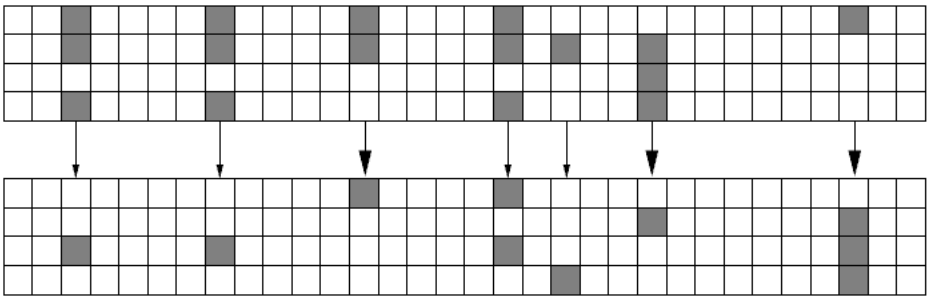
After the linear transformation and the application of S_1, we get the following approximation with bias 2^{-7}:



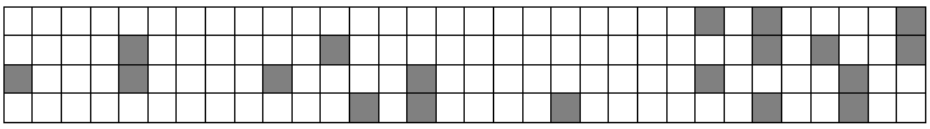
After the linear transformation and the application of S_2, we get the following approximation with bias 2^{-6}:



After the linear transformation and the application of S_3, we get the following approximation with bias 2^{-12}:



After the last linear transformation, we get the following output mask with 12 active S-boxes:



The total bias of the approximation is $2^8 \cdot 2^{-15} \cdot 2^{-7} \cdot 2^{-5} \cdot 2^{-3} \cdot 2^{-5} \cdot 2^{-6} \cdot 2^{-7} \cdot 2^{-6} \cdot 2^{-12} = 2^{-58}$.

References

1. R. Anderson, E. Biham, L. Knudsen, *Serpent: A Proposal for the Advanced Encryption Standard*, in the proceedings of the First Advanced Encryption Standard (AES) Conference, Ventura, CA, 1998.
2. E. Biham, O. Dunkelman, N. Keller, *Linear Cryptanalysis of Reduced Round Serpent*, in the Proceedings of FSE 2001, Lecture Notes in Computer Science, vol. 2355, pp. 16-27, Yokohama, Japan, April 2001.
3. E. Biham, O.Dunkelman, N.Keller, *The Rectangle Attack - Rectangling the Serpent*, Advances in Cryptology - Eurocrypt'01 (Lecture Notes in Computer Science no.2045), pp. 340-357, Springer-Verlag, 2001.
4. T. Kohno, J.Kelsey, B.Schneier, *Preliminary Cryptanalysis of Reduced-Round Serpent*, AES Candidate Conference, pp. 195-211, 2000