AES MixColumn with 94 XOR gates

Alexander Maximov

Ericsson Research, Lund, Sweden
alexander.maximov@ericsson.com

Abstract. In this short report we present a short linear program for AES MixColumn with 94 XOR gates.

Keywords: AES · MixColumn · Short Linear Program

1 Introduction

The part MixColumn of AES encryption round, applied to the AES state $\{r_{i,j}\}$ for $0 \leq i, j \leq 3$, is the following column-wise matrix multiplication.

$$\begin{bmatrix} r'_{0,j} \\ r'_{1,j} \\ r'_{2,j} \\ r'_{3,j} \end{bmatrix} = \begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix} \cdot \begin{bmatrix} r_{0,j} \\ r_{1,j} \\ r_{2,j} \\ r_{3,j} \end{bmatrix}, 0 \leq j \leq 3.$$

The classical circuit of MixColumn needs 108 2-input XOR gates and can be implemented as follows: for each $0 \leq j \leq 3$ do: $t_0 = r_0 + r_1$, $t_1 = r_1 + r_2$, $t_2 = r_2 + r_3$, $t_3 = r_3 + r_0$, and then $r'_0 = 2t_0 + t_2 + r_1$, $r'_1 = 2t_1 + t_3 + r_2$, $r'_2 = 2t_2 + t_0 + r_3$, $r'_3 = 2t_3 + t_1 + r_0$, where multiplication $2t_i$ is the multiplication by x in the Rijndael field and can be implemented with three 2-input XOR gates.

Previous results. There were several improvements to the classical circuit. At CHES 2017 a new circuit of MixColum with 103 XOR gates was presented in [JMPS17]. Later on, an improved version with 97 gates was given in [KLSW17]. Recently, there was a new paper published on IACR ePrint [EJMY19] where in Appendix F the authors presented MixColumn with 95 gates. We have also received the information that a paper presenting a circuit for MixColumn with 95 gates is accepted in the conference IWSEC-2019, which will be held on August 28-30, 2019.

Our results. In this short report we present a short linear program for AES MixColumn with 94 gates. At our best knowledge it is the smallest as of today.

2 Results

This work was mainly based on the parts of the algorithm given by Boyar et al [BP10], as well as our own techniques presented in [EM19]. We wrote a search program that combines Boyar's algorithm to compute the shortest distance, and our ideas for metrics and the search tree. In our simulations we used 5000 leaves of the search tree with 150 leaves being extended from each leaf.

Surprisingly, the best result was achieved when we tried to *minimize* the Euclidean norm metric. As it was mentioned in [EM19], the norm metric (ν) is not stable. When the norm is maximized, the algorithm tends to accept a gate that reduces distances (δ_i) to targets "unevenly", i.e., it is a greedy approach. When the norm is minimized, the distances are reducing more evenly, thus giving more chances for shared gates on the final steps of the search. However, which approach is better (to maximize or to minimize the norm) is still unclear – for different input matrices different approaches work better.

In the circuit below, x is the 32-bit input value, and y is the 32-bit output value.

```
t0 = x15 ^ x23
t1 = x7 ^ x31
t2 = x23 ^ x31
t3 = x7 ^ x15
t4 = x6 ^ x14
t5 = x4 ^ x12
t6 = x3 ^ x11
t7 = x0 ^ x8
t8 = x13 ^ x21
t9 = x5 ^ x29
t10 = x20 ^ x28
t11 = x16 ^ x24
t12 = x19 ^ x27
t13 = x22 ^ x30
t14 = x17 ^ x25
t15 = x1 ^ x9
t16 = x10 ^ x18
t17 = x2 ^ x26
t18 = x24 ^ t7
y16 = t2 ^ t18
t19 = t1 ^ t18
y24 = t11 ^ t19
t20 = x0 ^ t11
y8 = t0 ^ t20
t21 = t3 ^ t7
y0 = t20 ^ t21
t22 = x22 ^ t4
y30 = t9 ^ t22
t23 = x6 ^ x7
t24 = x5 ^ t13
t25 = x13 ^ t10
y21 = t9 ^ t25
t26 = x26 ^ t16
y2 = t15 ^ t26
t27 = x9 ^ t17
t28 = x28 ^ t8
t29 = x25 ^ y2
y26 = t27 ^ t29
t30 = t2 ^ t26
t31 = x3 ^ t0
t32 = x19 ^ t6
t33 = t1 ^ t32
y27 = t17 ^ t33
t34 = t5 ^ t12
t35 = x27 ^ t30
t36 = x10 ^ t35
y19 = t6 ^ t36
t37 = t16 ^ t31
y11 = t12 ^ t37
t38 = t14 ^ y8
t39 = t18 ^ t38
y9 = x1 ^ t39
t40 = t29 ^ t30
y17 = t11 ^ t40
t41 = x14 ^ t24
y6 = x13 ^ t41
t42 = x29 ^ t8
y5 = t5 ^ t42
t43 = x30 ^ t23
y31 = t0 ^ t43
t44 = t2 ^ t4
y7 = x15 ^ t44
t45 = x28 ^ t34
y20 = t2 ^ t45
t46 = t1 ^ t13
y23 = x15 ^ t46
t47 = y27 ^ t37
y3 = t36 ^ t47
t48 = t14 ^ t17
y18 = x10 ^ t48
t49 = x6 ^ t8
y14 = t13 ^ t49
t50 = x21 ^ y30
y22 = t24 ^ t50
t51 = x4 ^ x5
y29 = t28 ^ t51
t52 = x22 ^ t23
y15 = t44 ^ t52
t53 = t39 ^ y17
y25 = t21 ^ t53
t54 = x17 ^ t27
y10 = x18 ^ t54
t55 = x9 ^ t21
y1 = t14 ^ t55
t56 = x12 ^ y21
y13 = t28 ^ t56
t57 = t3 ^ t10
t58 = t6 ^ t57
y4 = x12 ^ t58
t59 = t31 ^ t32
t60 = x4 ^ t10
y12 = t59 ^ t60
t61 = y20 ^ t58
y28 = t59 ^ t61
```

Listing 1: MixColumn with 94 gates

The analysis of the above circuit revealed that in the final steps of the search, when all distances are $\delta_i \leq 2$, there are 2 targets having 2 ending solutions each, involving different intermediate gates. This indicates that the found circuit might have a redundancy with 2 gates and, therefore, we can make the following conjecture.

Conjecture 1. *We believe there exists a circuit for MixColumn with 92 XOR gates.*

References

[BP10] Joan Boyar and René Peralta. A New Combinational Logic Minimization Technique with Applications to Cryptology. In Paola Festa, editor, *Experimental Algorithms*, pages 178–189, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

[EJMY19] Patrik Ekdahl, Thomas Johansson, Alexander Maximov, and Jing Yang. A new SNOW stream cipher called SNOW-V. Cryptology ePrint Archive, Report 2018/1143 version 20190603:135044, June 3, 2019. https://eprint.iacr.org/eprint-bin/getfile.pl?entry=2018/1143&version=20190603:135044&file=1143.pdf.

[EM19] Patrik Ekdahl and Alexander Maximov. New Circuit Minimization Techniques for Smaller and Faster AES SBoxes. Cryptology ePrint Archive, Report 2019/802, 2019. https://eprint.iacr.org/2019/802.

[JMPS17] Jmy Jean, Amir Moradi, Thomas Peyrin, and Pascal Sasdrich. Bit-Sliding: A Generic Technique for Bit-Serial Implementations of SPN-based Primitives. In *Cryptographic Hardware and Embedded Systems CHES 2017*, volume 10529 of *Lecture Notes in Computer Science*, pages 687–707. Springer, 2017.

[KLSW17] Thorsten Kranz, Gregor Leander, Ko Stoffelen, and Friedrich Wiemer. Shorter Linear Straight-Line Programs for MDS Matrices. *IACR Transactions on Symmetric Cryptology*, 2017(4):188–211, Dec. 2017.