SCIENCE CHINA
Information Sciences

• LETTER •

August 2021, Vol. 64 189402:1–189402:2
https://doi.org/10.1007/s11432-020-3089-9

Fast substitution-box evaluation algorithm and its efficient masking scheme for block ciphers

Hai HUANG[1,2], Leibo LIU[1*], Min ZHU[1], Shouyi YIN[1] & Shaojun WEI[1]

[1]*Institute of Microelectronics, Tsinghua University, Beijing* 100084*, China;*
[2]*Harbin University of Science and Technology, Harbin* 150080*, China*

Received 23 April 2020/Revised 6 July 2020/Accepted 10 October 2020/Published online 21 May 2021

Citation Huang H, Liu L B, Zhu M, et al. Fast substitution-box evaluation algorithm and its efficient masking scheme for block ciphers. Sci China Inf Sci, 2021, 64(8): 189402, https://doi.org/10.1007/s11432-020-3089-9

Dear editor,

Higher-order masking is one of the most effective countermeasures against differential power analysis attacks (DPA) [1]. However, this measure has not yet been widely applied to real-life applications because it suffers from heavy masking-overheads. The efficient realization of non-linear operations, such as substitution-box (S-box), is a major challenge. Among them, the addition-chain(AC)-based schemes are the most compelling approaches for addressing the efficiency issue. These schemes provide a feasible solution by reducing the number of non-linear multiplication required for the power function [2]. However, it is very hard to further reduce the masking complexity by existing methods.

This study proposes a fast S-box evaluation algorithm using a look-up-table based addition-chain (LUT-AC), which is inspired by AC based algorithm. This algorithm substantially reduces the masking-complexity by replacing a certain power function with a specially designed LUT. Its corresponding higher-order masking scheme derived from the LUT-AC is also proposed; it provides a new practical way for masking S-box of block ciphers.

Proposed LUT-AC. The LUT-AC is a new type of addition chain that is constructed by mixing LUT with a normal AC. In the LUT-AC, the objective power function is separated into a certain power function and a specially designed LUT, which is shown in Figure 1 (take x^{254} as an example). For the masking scheme, if the masking-complexity is to be further reduced, then replacing the specific power function (usually close to the objective power function, e.g., x^{238}) with a small size LUT will offer a possible solution, i.e., the LUT-AC based masking scheme. Because it can reduce both the non-linear multiplication number and the linear operation number.

Fast algorithm for solving the high-degree congruence equation over GF(2^n). The elements over GF(2^n) are obtained through the function of $\mathbb{F}_2 \to (\mathbb{F}_2/p(x), \oplus, \otimes)$. There are 2^n congruence classes with regard to fixed p, e.g., there are 2^8 elements over GF(2^8) [3]. If $mk = 2^n - 1$, m and k are natural numbers, then all the elements over GF(2^8) satisfy

$$x^m \equiv b_j (\mathrm{mod}\ p), \quad j \in [1, k]. \tag{1}$$

Theorem 1. If a is the primitive root modulo p over GF(2^n), then the solutions of $x^m \equiv 1(\mathrm{mod}\ p)$ satisfy

$$x_j = a^{(j-1)^k}, \quad k = \frac{2^n - 1}{m}, \quad j \in [1, m]\,. \tag{2}$$

Theorem 2. For $mk = 2^n - 1$, if all solutions of $x^m \equiv 1(\mathrm{mod}\ p)$ is the vector X_1, and a is one of the primitive roots, then the solutions of other $k-1$ high-degree congruence equations can be easily obtained from

$$(a^{(j-1)} \otimes X_0)^m \equiv b_j (\mathrm{mod}\ p), \quad j \in [2, k], \tag{3}$$

where b_j is one solution of $x^k \equiv 1(\mathrm{mod}\ p)$.

Proofs of Theorems 1 and 2 can be found in Appendixes A and B.

Based on the above theorems, a fast algorithm can be developed to solve some specific-degree congruence equations which satisfy $(x^m)^d \equiv b_j(\mathrm{mod}\ p)$, where m is one factor of $2^n - 1$, $d \in \mathbb{N}$ and $j \in [1, k]$. Since all elements over GF(2^n) can be partitioned into m sets with all k elements in each set having identical power functions. As a result of this property, the LUT used to evaluate the specific power function can be efficiently constructed. For GF(2^n), if $2^n - 1$ is a composite number, then it will have different factorizations with different factoring pairs (p_i, q_i) that satisfy $2^n - 1 = p_i \times q_i$, $i \in \mathbb{N}$. Take the factor pair (m, k) as an example, now, equation $x^m \equiv 1(\mathrm{mod}\ p)$ can be solved under Theorem 1. Next, the other $k-1$ equations can be solved under Theorem 2. Finally, the solutions of $(x^m)^d \equiv b_j(\mathrm{mod}\ p)$ can be obtained by raising the power of b_j to d. Appendix C lists the high-degree congruence solutions with different factoring pairs for GF(2^n) where $n \leqslant 9$. As we have known, the size of the S-boxes in a cryptosystem is typically less than nine.

Masking scheme for AES S-box. For AES S-box evaluation, different LUT-ACs can be selected according to the requirements of real-life applications. The LUT size is dependent on the factoring pairs; as such, it is less than the

* Corresponding author (email: liulb@tsinghua.edu.cn)

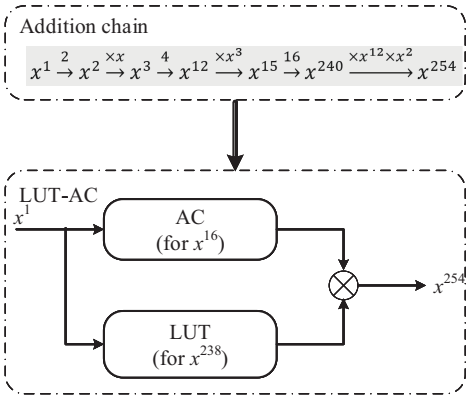

Figure 1 Proposed LUT-AC.

square root of the original size (i.e., 256 bytes) in most situations. If the memory storage is not a constrained resource, then the AC with large size LUT (e.g., $x^{16} \xrightarrow{\times x^{238}} x^{254}$, which requires 15 bytes of memory) can be selected; while, for constrained resources, the AC with small size LUT (e.g., $x^{16} \xrightarrow{\times x^{85} \times x^{153}} x^{254}$, which requires eight bytes memory) can be selected. Therefore, the LUT-AC based scheme has high flexibility to meet different application requirements.

To mask AES S-box, there are two non-linear parts: one non-linear multiplication and one small size LUT, which are required to be masked separately. For the non-linear multiplication, a secure ISW [4] scheme can be used. To mask the LUT, the multiplicative masking is easier and more efficient than the Boolean masking because the LUT is constructed based on the higher-order congruence equations.

The high-degree congruence equations and their solutions which used to construct the AES S-box, can be viewed as the extension matrix E:

$$E = [B|M], \tag{4}$$

where $B = [b_1 \cdots b_k]_{k\times 1}$, $b_j = a^{(k-1)m}$, $j \in [1, \frac{2^n}{m}]$, and M is the solutions matrix.

According to Theorem 3 in [5], the mixed masking scheme (Boolean masking for linear operation and multiplicative masking for the LUT) is developed. Since the proposed algorithm contains both Boolean and multiplicative masking schemes, the masks have to be converted from one type to the other. Thus, the transformation from Boolean masking to multiplicative masking (BMtoBM) shown in (5) is required.

$$\begin{aligned}&(x = \oplus_{i=0}^{d} xa_0, xa_1, \ldots, xa_d)\\ &\rightarrow (x = \otimes_{i=0}^{d} xm_0, xm_1, \ldots, xm_d).\end{aligned} \tag{5}$$

In order to simplify the operation and fully utilize the properties of the proposed algorithm, the 'BMtoMM' is integrated with LUT and denoted as a global LUT (GLUT). Therefore, the LUT integrated with masking conversion can be masked as follows. First, the Boolean masks of the inputs are transformed to multiplicative masks; then, the multiplicative masks are unmasked by table re-computation (i.e., permutation of b_j); finally, the outputs are accessed from LUT and remasked by new random masks. The detailed algorithm is shown in Appendix D.

Masking-overheads comparisons. To evaluate the masking-overheads of the proposed schemes, the computing-complexity for AES S-box is evaluated and compared with existing AC-based schemes in [6–9]. Under a normal AC-based S-box evaluation schemes, there are three types of arithmetic modules: F_2-linear operation, multiplication, and $x \times g(x)$ over GF(2^n). Unlike these existing schemes, the proposed scheme included a GLUT and a table re-computation operation. The computing-complexity of the different arithmetic modules costs and different AES S-boxes is shown in Appendix E.

Conclusion. The computing-complexity (in terms of XOR, non-linear multiplication over GF(2^n), and LUT access) is reduced by approximately 62.5% compared with the existing schemes. Therefore, the proposed scheme has the lowest complexity compared to existing higher-order masking schemes. Furthermore, the proposed algorithm is a general method to evaluate the inverse over GF(2^n), and thus it is suitable for every block cipher constructed with inverse functions and affine transformations, e.g., SM4 and Camellia.

Acknowledgements This work was supported by Optoelectronics and Microelectronic Devices and Integration of National Key R&D Program of China (Grant No. 2018YFB2202100) and Heilongjiang Provincial Natural Science Foundation of China (Grant No. YQ2019F010).

Supporting information Appendixes A–E. The supporting information is available online at info.scichina.com and link.springer.com. The supporting materials are published as submitted, without typesetting or editing. The responsibility for scientific accuracy and content remains entirely with the authors.

References

1 Kocher P, Jaffe J, Jun B. Differential power analysis. In: Proceedings of Annual International Cryptology Conference, 1999. 388–397
2 Carlet C, Prouff E. Polynomial evaluation and side channel analysis. In: The New Codebreakers. Berlin: Springer 2016. 315–341
3 Childs L N. Part II congruence classes and rings. In: A Concrete Introduction to Higher Algebra. 3rd ed. Berlin: Springer, 2009. 93–123
4 Ishai Y, Sahai A, Wagner D. Private circuits, securing hardware against probing attacks. In: Proceedings of the 23rd Annual International Cryptology Conference, 2003. 463–481
5 Huang H, Liu L L, Huang Q H, et al. Low area-overhead low-entropy masking scheme (LEMS) against correlation power analysis attack. IEEE Trans Comput-Aided Des Integr Circ Syst, 2019, 38: 208–219
6 Grosso V, Prouff E, Standaert F-X. Efficient masked s-boxes processing a step forward. In: Proceedings of the 7th International Conference on Cryptology in Africa, 2014. 251–266
7 Rivain M, Prouff E. Provably secure higher-order masking of AES. In: Proceedings of Workshop Cryptographic Hardware and Embedded Systems (CHES'10), 2010. 413–427
8 Carlet C, Goubin L, Prouff E, et al. Higher-order masking schemes for S-Boxes. In: Proceedings of Workshop Fast Software Encryption (FSE'12), 2012. 366–384
9 Coron J-S, Kizhvatov I, Roy A, et al. Analysis and improvement of the generic higher-order masking scheme of FSE 2012. In: Proceedings of Workshop Cryptographic Hardware and Embedded Systems (CHES'13), 2013. 417–434