INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2024, VOL. 71, NO. 2, PP. 477-482
Manuscript received December 5,2024; revised May, 2025. doi: 10.24425/ijet.2025.153594

Nonlinear degree of Ascon permutation

Victor Ruzhentsev

***Abstract*—An estimation of the nonlinear degrees for the forward and inverse permutations of the Ascon algorithm is made in this work. This estimation is made by analyzing higher order differentials.**

The obtained results of nonlinear degree are significantly lower than the known data. Instead of the generally accepted values s^r (where s is nonlinear degree of substitution and r is number of rounds), the computational experiments demonstrated the value $s^{(r-1)}+1$ in all the considered cases.

These results allow to clarify the complexity of constructing the best known distinguisher - the zero-sum distinguisher - for a multi-round transformations. Thus, instead of the known complexity values of 2^{85} and 2^{130} for 11 and 12 rounds of transformations, according to our data, the complexity for 11 rounds is 2^{35} and for 12 rounds is 2^{70}.

***Keywords*—Permutation of Ascon; Nonlinear degree; Nonlinear degree of substitution; Zero-sum distinguisher**

I. INTRODUCTION

THE Ascon algorithm was announced as the winner of the Lightweight Cryptography competition [1] in 2023. Therefore, today much attention of the world cryptographic community is focused on the analysis of this algorithm.

Conventionally, work on the cryptographic analysis of this algorithm can be divided into several directions, one of which is related to the study of the properties of the multi-round permutation p. This p permutation, using the key, associated information, and other input information, works in a stream mode to produce a bit sequence that will be XORed with the plaintext.

The best distinguishing attacks on this permutation p - a zero-sum attack [2] or best known key recovery attack - cube attack [3] - take advantage of the fact that this transformation p uses nonlinear substitutions with a low nonlinear degree (nonlinear degree of substitution is 2, nonlinear degree of inverse substitution is 3), and therefore also has a low nonlinear degree even after a large number of rounds. As a result, in accordance with [4], all differentials of degree d for a transformation with degree of nonlinearity d will be equal to each other, and differentials of degree d+1, accordingly, will be equal to 0.

However, it is difficult to determine accurately the degree of nonlinearity of the multiround transformation p. Well-known approaches make it possible to determine the upper bound of the degree of nonlinearity based on the nonlinear degree of substitutions s and the number of rounds r. However, the resulting value s^r may be greatly overestimated, and as a result, the actual security against the distinguishing attack may be significantly lower than expected.

In [5], Theorems were proved, which clarify the resulting estimate of the degree of nonlinearity. However, these results work for a large number of rounds when nonlinear degree is close to the threshold value n-1 for an n-bit block.

II. OBJECTIVES

In this work, an attempt is made, using the method proposed in [4], to estimate the nonlinear degrees for the forward and inverse permutations of the Ascon algorithm by analyzing higher order differentials. Such refined estimations also make it possible to clarify the complexity of the best known distinguishing attacks for Ascon permutation - Zero-sum attack [2].

III. THE ASCON ALGORITHM

A. General information

The Ascon was developed in 2014 by a team of researchers C. Dobraunig, M. Eichlseder, F. Mendel, M. Schläffer from Graz University of Technology, Infineon Technologies, Lamarr Security Research, and Radboud University. The cipher family was chosen as a finalist of the CAESAR Competition in February 2019. The Ascon [7] also had been selected by US National Institute of Standards and Technology (NIST) for future standardization of the lightweight cryptography in 2023.

The algorithm uses components that have already been proven by use in other well-known cryptographic algorithms. Sponge construction and S-box are from Keccak (SHA-3)[5], linear layer is from SHA-2 [8].

B. Modes of operation and Sponge schemes of Ascon

There are two main modes of operation of this algorithm: AEAD mode (Authenticated Encryption with Associated Data) and Hash mode.

The scheme of Sponge construction of Ascon for AEAD mode from [7] is presented on Fig. 1. The sponge scheme is one of the most popular schemes among the algorithms-participants of the Lightweight Cryptography competition [1] (16 of 32 participants of Round 2 competition used Sponge scheme).

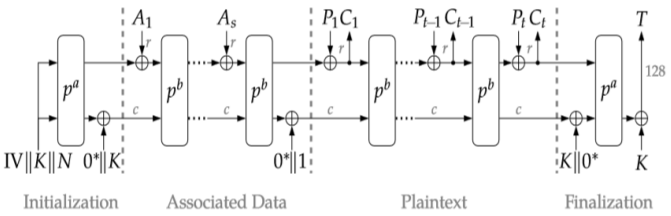

Fig. 1. Sponge construction of Ascon in AEAD mode [7]

Author is with Kharkiv National University of Radioelectronics
(e-mail: viktor.ruzhentsev@nure.ua).

At the initialization stage (part “Initialization” in the Fig. 1), a 320-bit internal state is formed using a 128-bit key K, an initial vector IV, 128-bit nonce N, and a 12-round transformation p^a. Next, the "absorption" of the associated data (“Associated data” in the Fig. 1) already takes place using a 6-round or 8-round transformation p^b. Using the same transformation p^b, 64-bits or 128-bits blocks of ciphertext C_i are formed after the encryption of corresponding size plaintext blocks P_i (part “Plaintext” in the Fig. 1). This process is similar to the stream mode, since each block of the cryptogram is formed as a result of XOR adding a block of plaintext and bits of the current internal state. The 128-bits tag T is formed using the 12-round transformation p^a and the key K in the final part (“Finalization” in the Fig. 1). The purpose of tag T is to ensure the integrity of the message.

Decryption is performed according to the same scheme, except that instead of plaintext blocks P_i, cryptogram blocks C_i will be added by XOR, and plaintext blocks will be formed.

The sponge scheme for Ascon in Hash modes is shown in Fig. 2.

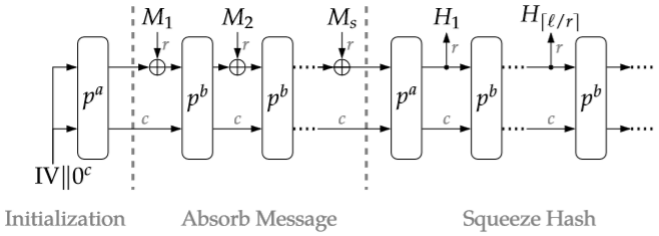

Fig. 2. Sponge construction of Ascon in Hash mode [7]

At the initialization stage (part “Initialization” in the Fig. 2), a 320-bit internal state is formed using an initial vector IV and a 12-round transformation p^a. Next, the "absorption" of the 64-bits blocks of message (part “Absorb Message” in the Fig. 2) takes place using a 8-round or 12-round transformation p^b. Using the same transformation p^b, blocks of hash-code H_i are formed (part “Squeeze Hash” in the Fig. 2). Size of hash-code is 256 bits.

The main advantages of Sponge scheme are its versatility and the ability to be tuned to achieve good performance in any domain, including high-speed implementation, memory-constrained environments, and regular desktop computers.

The presence of such an external Sponge scheme allows, by adjusting the parameters c and r (see Fig. 1, 2), to find a certain compromise between speed and security, leaving the internal transformations unchanged.

C. Internal permutation p

The key to the success of such a scheme is also an efficient internal transformation p. This transformation p uses proven elements. In addition to the constant addition operation, these are the oriented on efficient implementation 5-to-5 bit substitution (taken from the Keccak hashing algorithm) and a linear transformation that uses rotations and XOR-additions (very similar to the transformations from the SHA-2 [8], similar linear transformations were also used in the Noekeon algorithm [9]).

From the point of view of efficient implementation, it is important that linear transformations can be performed by operating on five 64-bit blocks which form the 320-bit internal state. If we denote the five input 64-bit blocks as {X0, X1, X2, X3, X4}, and the output blocks as {Y0, Y1, Y2, Y3, Y4}, then the linear transformation is performed as follows:

Y0 = X0 + (X0 >>> 19) + (X0 >>> 28);
Y1 = X1 + (X1 >>> 39) + (X1 >>> 61);
Y2 = X2 + (X2 >>> 1) + (X2 >>> 6);
Y3 = X3 + (X3 >>> 10) + (X3 >>> 17);
Y4 = X4 + (X4 >>> 7) + (X4 >>> 41).

As can be seen, the linear transformation can be effectively implemented on a wide range of computing platforms. From the point of view security, it is stated that the branch number is 4.

Table I shows a 5-to-5 bit substitution that must be performed 64 times for a 320-bit block.

TABLE I
SUBSTITUTION TABLE 5-TO-5 BITS

X	0	1	2	3	4	5	6	7
S(X)	4	B	1F	14	1A	15	9	2
X	8	9	A	B	C	D	E	F
S(X)	1B	5	8	12	1D	3	6	1C
X	10	11	12	13	14	15	16	17
S(X)	1E	13	7	E	0	D	11	18
X	18	19	1A	1B	1C	1D	1E	1F
S(X)	10	C	1	19	16	A	F	17

The efficiency of this transformation is that these 64 substitutions for the entire 320-bit block can be performed using only 22 logical operations on five 64-bit subblocks [7]. The scheme of these logical operations is shown in Fig. 3.

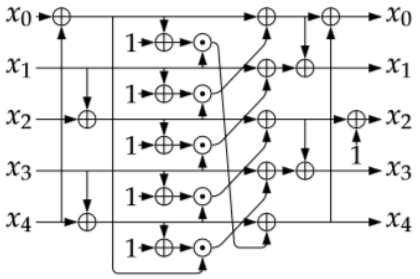

Fig. 3. 5-to-5 bit substitution using 22 logical operations on five 64-bit subblocks

In addition to the efficient implementation in [7], the following properties are claimed for the substitution:

- Invertible and no fix-points,
- Each output bit depends on at least 4 input bits,
-Algebraic degree 2,
- Maximum differential probability and linear bias 1/4,
- Differential and linear branch number 3.

Thus, in addition to the usual requirements of ensuring good mixing and a certain level of nonlinearity, the substitution also partially solves the problem of ensuring dispersion.

D. Inverse internal permutation p^{-1}

Normal operation of the Ascon algorithm does not use the inverse transformation, but as part of the research, it is necessary to analyze and determine the nonlinear degree for the inverse internal permutation p^{-1}.

Materials from [10] were used to implement the inverse permutation p^{-1}. Since the normal operation of the Ascon algorithm does not use the inverse transformation, it is not aimed at fast and efficient implementation. In general, it must be performed the inverse transformations in the reverse order to get the inverse transformation.

The inverse linear transformation, like the direct linear transformation (see above), contains the operations of rotation and XOR-addition, but significantly more number of operations than the direct linear transformation. When performing the inverse linear transformation, each of the five 64-bit blocks requires from 31 to 35 rotations and the same number of XOR-addition operations (for example, for the first 64-bit block - 31 rotations and the same number of XOR-addition operations: rotations by 0, 3, 6, 9, 11, 12, 14, 15, 17, 18, 19, 21, 22, 24, 25, 27, 30, 33, 36, 38, 39, 41, 42, 44, 45, 47, 50, 53, 57, 60, 63 bits, and then adding the results by XOR). Thus, to get inverse operation it must be used for about 10 times more operations than for direct linear transformations.

The inverse substitution of 5-to-5 bits must be done 64 times for 320-bit block. There is no way to replace these substitutions with logical operations on 64-bit blocks. Compared to the 22 fast logical operations for forward substitution, implementation of these inverse substitutions will also be several times slower.

Thus, software implementation of inverse round transformation is much slower than direct transformation (the difference in speed is more than 10 times).

IV. Review of security analysis results for Ascon permutation

One of the directions of algorithm security analysis is related to the search for distinguishing attacks on the multi-round internal transformation p. Such distinguishing attacks do not indicate the direct vulnerability of the algorithm, but subsequently they can lead to the other more dangerous attacks, the purpose of which will be to find a secret key or to forge a message for which integrity must be ensured.

The work [7] provides an overview of the best distinguishing attacks on the permutation p of the Ascon algorithm. The maximum number of rounds that can be distinguished from a random transformation for different cryptanalytic methods are: Zero-sum attack – 20, Integral attack - 11, Differential attack - 5, Linear attack - 5, Impossible differential - 5. As can be seen, the zero-sum attack is significantly more effective than the other distinguishing attacks.

During the Zero-sum attack [2], a set of input and corresponding output blocks is formed for permutation p, the sum of these input blocks is equal to 0 and the sum of corresponding output blocks is also equal to 0. The Zero-sum attack on the Ascon algorithm uses a low nonlinear degree of the used inside permutation p substitutions. As a result, all differentials of degree d+1 for a function with a nonlinear degree d will be equal to 0.

In [7], the complexity of constructing a zero-sum distinguisher for 11 and 12 rounds is estimated to 2^{85} and 2^{130}, respectively. The distinguishing Zero-sum attack can be extended to more rounds (up to 20 rounds) [2].

These estimates are obtained based on the nonlinear degree for multi-round transformations, therefore, refining the nonlinear degree for direct and inverse transformations will also allow us to refine the complexity of Zero-sum attacks.

V. Methods for estimating nonlinear degree of encryption scheme

A. Standard (classical) approach

Well-known approaches make it possible to determine the upper bound of the nonlinear degree based on the nonlinear degree of substitutions s and the number of rounds r. However, the resulting value s^r may be greatly overestimated, and as a result, the actual security against some attack may be significantly lower than expected.

Upper bounds of algebraic degree after r rounds of Ascon permutation from [7] are presented in Table II.

TABLE II
UPPER BOUND OF ALGEBRAIC DEGREE AFTER r ROUNDS OF ASCON PERMUTATION [7]

Number of rounds, r	Nonlinear degree for direct permutation	Nonlinear degree for inverse permutation
1	2	3
2	4	9
3	8	27
4	16	81
5	32	209
6	64	283
7	128	307
8	256	314
9	298	318
10	312	319
11	317	
12	319	

Using this standard approach, an assessment of the nonlinear degree was made for rounds 1-8 of the direct transformation (s = 2) and for rounds 1-4 of the inverse transformation (s = 3) and the results are presented in Table II (corresponding cells of table are highlighted by grey color).

Values for other cells are formed with using Theorems from [6], which clarify the resulting estimate of the nonlinear degree. However, these Theorems work for a large number of rounds when degree of nonlinearity is close to the threshold value n-1 for an n-bit block.

B. Knudsen's method

In [4] a method for estimating the nonlinear degree of an encryption transformation based on an analysis of the values of higher-order differentials was proposed. It is known that all differentials of degree d for a transformation with degree of nonlinearity d will be equal to each other, and differentials of degree d+1 will, accordingly, be equal to 0.

Test for nonlinear order from [4] is presented on Fig. 4.

Input: E_k – block cipher, k – key, b – the boundary order of the differential that can be calculated in a reasonable time, plaintexts $p1$!= $p2$.

Output: i <= b – minimum nonlinear order (degree) of E_k .

Let $a1, a2, \ldots, ai$ be linearly independent.
1) Set i=1.
2) Compute differentials of order i
$y1=\text{dif}_i(a1, a2, \ldots, ai)\ E_k(p1)$ and
$y2=\text{dif}_i(a1, a2, \ldots, ai)\ E_k(p2)$.
3) If $y1 = y2$ output i and stop.
4) If $i >= b$ output i and stop.
5) Set $i = i+1$ and go to step 2.

Fig. 4. Test for nonlinear order [4]

For Test for nonlinear order to work, linearly independent difference values $a1, a2, \ldots, ai$ for different levels of differentials and an arbitrary input values $p1, p2$ ($p1$ is not equal to $p2$) must be selected. The nonlinear degree of the permutation can be determined by gradually increasing the degree of differentials and checking the equality of their values.

In the algorithm on Fig. 4 $\text{dif}_i(a1, a2, \ldots, ai)\ E_k(p1)$ denotes the value of the differential of degree i, for the transformation E_k, the input value $p1$, by the values of the difference $a1, a2, \ldots, ai$ at different levels of the differential. To calculate such differentials of high degrees, the recursion function dif_n was used (see Table III).

In Table III, the symbol "+" denotes the XOR operation.

In [4] it is also said that the equality of differentials of some order does not immediately mean that the nonlinear degree is the same as the order of differentials. There may be other keys and other input values for which the differentials are not equal. There are no keys in the case of the Ascon permutation, and during the experiments we will consider a large number of input values. We will also consider differentials of order d and the next order d+1; if they are equal, then we will make a conclusion about the nonlinear degree.

TABLE III
FUNCTION **dif_n** TO CALCULATE DIFFERENTIAL OF DEGREE n

Function **dif_n**
Inputs: E – block cipher or some transformation, plaintext p, differences for different levels of differential $a1, a2, \ldots, ai$, order of differential n. Outputs: $\text{dif}_n(a1, a2, \ldots, ai)\ E(p)$ – differential.

1) If (n ==1)
{
return
$\text{dif}_1(a1, a2, \ldots, ai)\ E(p) = \text{E}(p) + \text{E}(p + a1)$;
}

2) else
{
return $\text{dif}_n\ (a1, a2, \ldots, ai)\ E(p) =$
$\text{dif}_{n-1}(a1, a2, \ldots, ai)\ E(p) +$
$\text{dif}_{n-1}(a1, a2, \ldots, ai)\ E(p + a_{n-1})$;
}

VI. COMPUTATIONAL EXPERIMENTS TO DETERMINE THE NONLINEAR DEGREE OF MULTI-ROUND ASCON PERMUTATIONS

As it was said in section III, materials from [10] were used to implement the inverse permutation p^{-1}. Since the normal operation of the Ascon algorithm does not use the inverse transformation, it is not aimed at fast and efficient implementation, and therefore software implementation of inverse transformation is much slower than direct transformation (the difference in speed is more than 10 times).

Using the Knudsen's method (see previous section), experiments were performed to determine the nonlinear degree of the direct and inverse multiround permutations of the Ascon algorithm. The results are presented in Tables III and IV, respectively.

TABLE III
NONLINEAR DEGREE FOR MULTIROUND ASCON PERMUTATION p

Number of Rounds	Maximum nonlinear degree	Total number of experiments	Percent of experiments with maximum nonlinear degree
2	2	100000	100
3	5	100000	22
4	8	100000	77
5	17	1000	74
6	33	1	100

TABLE IV
NONLINEAR DEGREE FOR MULTIROUND INVERSE ASCON PERMUTATION p^{-1}

Number of Rounds	Maximum nonlinear degree	Total number of experiments	Percent of experiments with maximum nonlinear degree
2	4	100000	100
3	10	100000	100
4	28	1	100

Computational experiments have confirmed the possibility of estimating differentials for an nonlinear degree of slightly more than 30 using calculations of higher order differentials, as stated in the work [4]. On an average laptop, calculating the 33rd order differential took about 4.5 hours for 6 forward rounds and about 3 hours to calculate the 28th order differential for 4 inverse rounds (approximately 10 times slower inverse transformation also significantly slows down the calculation of high-degree differentials).

It seems problematic to make calculation on conventional computing equipment for a bigger number of rounds of forward and inverse transformations than are presented in the tables III, IV.

For a smaller number of rounds, 100,000 experiments were used (the values of nonlinear degree did not change with further increasing number of experiments). Each experiment used two randomly generated input values $p1$ and $p2$ (see algorithm on Fig. 4).

In some experiments, the degree of nonlinearity was less than the maximum, so the last column of the tables III, IV shows the percentage of experiments with the maximum value of the nonlinear degree. In other cases, the nonlinear degree is less by one.

The presented in the tables III, IV results clearly evident that the maximum degree of nonlinearity for r-round forward and inverse transformations of the Ascon algorithm is upper bound by $s^{(r-1)}+1$ (instead of the traditional value s^r), where s is the nonlinear degree of the substitution (nonlinear degree of Ascon substitution is s=2, nonlinear degree of inverse substitution is s=3).

That is, using upper bound by $s^{(r-1)}+1$ for 7 rounds of forward direction the value of the nonlinear degree expected to be 65, and for 5 inverse rounds - 82.

VII. The Clarified Complexity of Building Zero-sum Distinguisher

Using the algorithm from [7] it can be estimated the complexity of building Zero-sum distinguisher. For refined nonlinear degrees of multi-round direct and inverse permutations (see Tables III, IV), the complexity will be 2^{70} for 12 rounds (4 inverse rounds, free middle round, and 7 forward rounds), 2^{35} for 11 rounds (4 + 1 + 6 rounds, with the data complexity a multiple of the S-box size 5 for the free inner round). These values are significantly lower than those presented in [7]: 2^{130} for 12 rounds and 2^{85} for 11 rounds.

We have more confidence in the correctness of complexity estimations for building the Zero-sum distinguisher than in value of nonlinear degree because exactly this zero-sum property is considered during our computational experiments.

It is also possible to estimate for what number of rounds Zero-sum distinguisher can be constructed. To do this, we first continue tables III and IV for a larger number of rounds using values $s^{(r-1)}+1$ for nonlinear degree (nonlinear degree of Ascon substitution is s=2, nonlinear degree of inverse substitution is s=3). The results are in Tables V, VI.

Table V
Expected Nonlinear degree for multiround Ascon permutation p

Number of Rounds	Expected nonlinear degree
7	65
8	125
9	257

Table VI
Expected Nonlinear degree for multiround inverse Ascon permutation p^{-1}

Number of Rounds	Expected nonlinear degree
5	82
6	244

Using the data from Tables V and VI, it can be estimated the complexity of constructing a Zero-sum distinguisher for the 16-round Ascon transform in 2^{260} (6 inverse rounds, free middle round, and 9 forward rounds).

VIII. Possible Ways of Theoretical Consideration of the Nonlinear Degree for the Ascon Permutation

The nonlinear degree of the multiround Ascon permutation is an important parameter that determines the complexity of potential attacks on this algorithm. Therefore, it is desirable to have a theoretical information about this parameter.

The type and parameters of linear transformations, in our opinion, do not have a significant effect on the degree of nonlinearity. To get such conclusion an additional computational experiments were performed. The dispersion of linear transformation were increased by using additional linear transformation from the Safer algorithm [11]. This transformation performs XOR-additions for five 64-bit subblocks X0, X1, X2, X3, X4 to produce five output 64-bit subblocks Y0, Y1, Y2, Y3, Y4 (see Fig. 5).

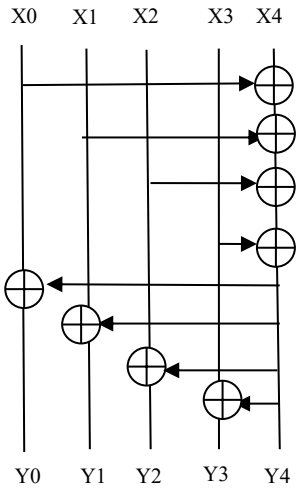

Figure 5. Additional linear transformation

This additional linear transformation did not lead to a change in the values of the nonlinear degree compared to the data presented in the Table III.

Possible ways of theoretical consideration of the nonlinear degree for the Ascon permutation, most likely, should be related to the analysis of substitution and its Boolean functions. During considering the nonlinear degree, it is probably necessary to consider what terms are in the Boolean functions and how, depending on this, the degree will increase with each additional round. This process may be similar to the analysis of terms presented in the Boolean functions when choosing cube variables in the cube attack [3].

Boolean functions for substitution of the Ascon algorithm and their analysis can be found in [3,7,10], but even a superficial look at Boolean functions allows to see some features. These are Boolean functions for a 5-bit column presented in [7]:

$$y_0 = x_4\, x_1 \oplus x_3 \oplus x_2\, x_1 \oplus x_2 \oplus x_1\, x_0 \oplus x_1 \oplus x_0,$$
$$y_1 = x_4 \oplus x_3\, x_2 \oplus x_3\, x_1 \oplus x_3 \oplus x_2\, x_1 \oplus x_2 \oplus x_1 \oplus x_0,$$
$$y_2 = x_4\, x_3 \oplus x_4 \oplus x_2 \oplus x_1 \oplus 1,$$
$$y_3 = x_4\, x_0 \oplus x_4 \oplus x_3\, x_0 \oplus x_3 \oplus x_2 \oplus x_1 \oplus x_0,$$
$$y_4 = x_4\, x_1 \oplus x_4 \oplus x_3 \oplus x_1\, x_0 \oplus x_1.$$

There are following features:

1) Some Boolean functions do not depend on all arguments (for example, y2 does not depend on x0).

2) A small number of terms of degree 2 (there are only 11 terms of degree 2 for 5 Boolean functions). Many input variables are absent in terms of degree 2. For example, the input variable $x2$ is present in terms of degree 2 only in Boolean functions $y0$ and $y1$. For the remaining Boolean functions, the dependence with x2 is linear.

3) Unequal participation of input variables in terms of degree 2. For example, variable $x1$ is involved in terms of degree 2 approximately two times more often than other input variables. As a consequence, with a fixed zero value of variable

$x1$, Boolean functions $y0$ and $y4$ do not contain terms of degree 2, i.e., these functions become linear.

CONCLUSIONS

1 The computational experiments made it possible to obtain more accurate estimations of the nonlinear degree for multiround direct and inverse permutations of the Ascon algorithm. The presented in Tables III, IV results clearly evident that the nonlinear degree for r-round forward (up to 6 rounds) and inverse (up to 4 rounds) transformations of the Ascon algorithm is upper bound by $s^{(r-1)}+1$ (instead of the traditional value s^r), where s is the nonlinear degree of the 5-to-5 bit substitution.

2 The computational experiments have shown that existing estimates of the nonlinear degree for the permutation of the Ascon algorithm are significantly overestimated. Lower values of nonlinear degree result in lower complexity of distinguishing and other attacks. For clarified nonlinear degrees of multiround direct and inverse transformations (see Tables III, IV), the complexity will be 2^{70} for 12 rounds (4 inverse rounds, free middle round, and 7 forward rounds), 2^{35} for 11 rounds (4 + 1 + 6 rounds). These values are significantly lower than those presented in [7]: 2^{130} for 12 rounds and 2^{85} for 11 rounds.

3 The nonlinear degree of the multiround Ascon permutation is an important parameter that determines the complexity of potential attacks. A promising direction of research seems to be a theoretical consideration of the nonlinear degree for the Ascon permutation to obtain more reasonable conclusions.

ACKNOWLEDGEMENTS

We are very grateful to the organizers and participants of the CECC 2024 conference for useful and fruitful communication.

REFERENCES

[1] Lightweight cryptography project of the American National Institute of Standards and Technology, 2015. https://csrc.nist.gov/projects/lightweight-cryptography

[2] Christina Boura and Anne Canteaut. Zero-sum distinguishers for iterated permutations and application to Keccak-f and Hamsi-256. In Alex Biryukov, Guang Gong, and Douglas R. Stinson, editors, Selected Areas in Cryptography, pages 1–17, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-19574-7

[3] Jules Baudrin, Anne Canteaut, and Léo Perrin. Practical cube attack against nonce misused Ascon. IACR Transactions on Symmetric Cryptology, 2022(4): pp. 120–144, Dec. 2022. https://doi.org/10.46586/tosc.v2022.i4.120-144

[4] Lars R. Knudsen. Truncated and higher order differentials. In Bart Preneel, editor, Fast Software Encryption, pp. 196–211, Berlin, Heidelberg, 1995. Springer Berlin Heidelberg. https://doi.org/10.1007/3-540-60590-8_16

[5] National Institute of Standards and Technology. DRAFT FIPS PUB 202, 2014.

[6] Christina Boura, Anne Canteaut, and Christophe De Cannière. Higher-order differential properties of Keccak and Luffa. In Antoine Joux, editor, Fast Software Encryption, pp. 252–269, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-21702-9_15

[7] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Martin Schläffer. Ascon v1.2. Submission to NIST, 2019. https://csrc.nist.gov/CSRC/media/Projects/lightweight-cryptography/documents/round-2/spec-doc-rnd2/ascon-spec-round2.pdf

[8] Draft, F.: Public comments on the draft federal information processing standard (fips) draft fips 180-2, secure hash standard (shs). http://dx.doi.org/10.6028/NIST.FIPS.180-4

[9] Daemen, J., Peeters, M., Van Assche, G.,Rijmen, V.: Nessie Proposal: NOEKEON. First Open NESSIE Workshop (2000), http://gro.noekeon.org

[10] Cihangir Tezcan. Truncated, impossible, and improbable differential analysis of ascon. Cryptology ePrint Archive, Paper 2016/490, 2016. https://eprint.iacr.org/2016/490

[11] J. L. Massey, G. H. Khachatrian, and M. K. Kuregian, “Nomination of SAFER++ as candidate algorithm for the New European Schemes for Signatures, Integrity, and Encryption (NESSIE).” Primitive submitted to NESSIE by Cylink Corp., Sept. 2000.