Rotational analysis of ChaCha permutation

Preprint · August 2020

CITATIONS
0

READS
140

3 authors:

Stefano Barbero
Università degli Studi di Torino
40 PUBLICATIONS **161** CITATIONS
SEE PROFILE

Emanuele Bellini
Technology Innovation Institute (TII)
65 PUBLICATIONS **293** CITATIONS
SEE PROFILE

Rusydi H. Makarim
19 PUBLICATIONS **89** CITATIONS
SEE PROFILE

Rotational analysis of ChaCha permutation

Stefano Barbero[1], Emanuele Bellini[2], and Rusydi Makarim[2]

[1] Politecnico di Torino, Italy
[2] Cryptography Research Centre, Technology Innovation Institute, UAE

Abstract. We show that the underlying permutation of ChaCha20 stream cipher does not behave as a random permutation for up to 17 rounds with respect to rotational cryptanalysis. In particular, we derive a lower and an upper bound for the rotational probability through ChaCha quarter round, we show how to extend the bound to a full round and then to the full permutation. The obtained bounds show that the probability to find what we call a parallel rotational collision is, for example, less than 2^{-488} for 17 rounds of ChaCha permutation, while for a random permutation of the same input size, this probability is 2^{-511}. We remark that our distinguisher is not an attack to ChaCha20 stream cipher, but rather a theoretical analysis of its internal permutation from the point of view of rotational cryptanalysis.

Keywords: ChaCha20 · Stream Cipher · Rotational cryptanalysis · Permutation · Distinguisher

Table of Contents

1 Introduction

Salsa20 [9] and ChaCha20 [8] are two closely related stream ciphers developed by Daniel J. Bernstein. Salsa20, the original cipher, was designed in 2005, then later submitted to the eSTREAM project by Daniel J. Bernstein [5]. Its detailed specification can be found in [5]. ChaCha20 is a modification of Salsa20, published by Bernstein in 2008, aimed at increasing diffusion and performance on some architectures. Google has selected ChaCha20 along with Bernsteins Poly1305 message authentication code as a replacement for RC4 in TLS, and its specifications can be found in [15]. Both ciphers are ARX (Add-Rotate-Xor) ciphers, i.e. built on a pseudorandom function based only on the following three operations: 32-bit modular addition, circular rotation, and bitwise exclusive or (XOR). This pseudorandom function is itself built upon a 512 bit permutation. According to [10], both permutations are not designed to simulate ideal permutations: they are designed to simulate ideal permutations with certain symmetries, i.e., ideal permutations of the orbits of the state space under these symmetries. The input of the Salsa and ChaCha function is partially fixed to specific asymmetric constants, guaranteeing that different inputs lie in different orbits. To our knowledge, while for Salsa some of these properties of "non-pseudorandomness" are well known, this is not the case for ChaCha (see Section subsection 1.3). Again to our knowledge, because of the use of asymmetric constants injected into the

input state of the permutation, none of these properties can be used to attack the entire stream cipher, or other ciphers where these permutations have been reused, as Salsa20 permutation in the Rumba20 compression function [7], a tweaked version of the ChaCha20 permutation in the BLAKE and BLAKE2 hash functions [1], or ChaCha12 permutation in the original SPHINCS post-quantum signature scheme [10] [3].

That said, studying mathematical properties of the Salsa and ChaCha permutations is still of theoretical interest, and it is useful to understand how these permutations can be reused to design other cryptographic primitives.

1.1 Our contribution

In this work, we show that ChaCha permutation does not behave as a random permutation, with respect to rotational cryptanalysis. To do so, we first derive and formally prove a lower and an upper bound for the probability of the propagation of rotational pairs through ChaCha quarter round. We provide experimental evidence of the correctness of the bounds by testing them on a toy version of ChaCha permutation. We then show how to extend the bounds to a full round and then to the full permutation. The obtained bounds allow us to distinguish ChaCha permutation, with for example 17 rounds, from a random permutation by using 2^{489} calls to an oracle running either ChaCha permutation or the random permutation. To do so, we prove that what we call a *parallel rotational collision*, is more likely to happen in ChaCha permutation, rather than in a random permutation. For example, such a collision happens with probability less than 2^{-488} for ChaCha permutation with 17 rounds, while with probability 2^{-511} for a random permutation. This distinguisher is *not* an attack to ChaCha20 stream cipher, but rather a theoretical analysis of its permutation from the point of view of rotational cryptanalysis.

1.2 Outline of the paper

In subsection 1.3 we briefly summarize the existing studies on the core function of Salsa and ChaCha stream ciphers. In section 2, we introduce the notation used throughout this manuscript and recall ChaCha permutation specifications. In section 3, we derive the lower and upper bound on the probability of the propagation of a rotational pair for ChaCha quarter round, for the full rounds, and for the full permutation. In section 4, we describe a distinguihser exploiting the above mentioned bounds. Finally, in section 5, we conclude the manuscript.

1.3 Related works

Often, rather than only considering the underlying permutation of Salsa and ChaCha, researchers study the so called Salsa (or ChaCha) *core* function (also

[3] The current SPHINCS submission to the NIST post-quantum standardization process does not use ChaCha anymore.

called ChaCha *block function* in [15]), whose output consists in applying the permutation and then xoring the output of the permutation with its input.

Already in the specifications of Salsa20 [6], there is an example showing how the 0 vector is a fixed point for Salsa core function. This is also true for ChaCha.

In [13], the authors find[4] an invariant for Salsa core main building block, the quarterround function, that is then extended to the row-round and column-round functions. This allows them to find an input subset of size 2^{32} for which the Salsa20 core behaves exactly as the transformation $f(x) = 2x$. This allows to construct 2^{31} collisions for any number of rounds. They also show a differential characteristic with probability one that proves that the Salsa20 core does not have 2nd preimage resistance. In [3], it is pointed out that none of the results in [13] has an impact on security of Salsa20 stream cipher, due to the use of fixed constants in the input. Indeed, Salsa20 is not designed to be a collision-resistant compression function [4].

In the Salsa20 security document [2, Section 4], two other symmetries of the cipher are reported, i.e.

- shifting the entire Salsa20 core input array along the diagonal has exactly the same effect on the output, i.e.

$$\begin{bmatrix} y_{0,0} & y_{0,1} & y_{0,2} & y_{0,3} \\ y_{1,0} & y_{1,1} & y_{1,2} & y_{1,3} \\ y_{2,0} & y_{2,1} & y_{2,2} & y_{2,3} \\ y_{3,0} & y_{3,1} & y_{3,2} & y_{3,3} \end{bmatrix} = \mathsf{Salsa}\left(\begin{bmatrix} x_{0,0} & x_{0,1} & x_{0,2} & x_{0,3} \\ x_{1,0} & x_{1,1} & x_{1,2} & x_{1,3} \\ x_{2,0} & x_{2,1} & x_{2,2} & x_{2,3} \\ x_{3,0} & x_{3,1} & x_{3,2} & x_{3,3} \end{bmatrix} \right)$$

$$\begin{bmatrix} y_{3,3} & y_{3,0} & y_{3,1} & y_{3,2} \\ y_{0,3} & y_{0,0} & y_{0,1} & y_{0,2} \\ y_{1,3} & y_{1,0} & y_{1,1} & y_{1,2} \\ y_{2,3} & y_{2,0} & y_{2,1} & y_{2,2} \end{bmatrix} = \mathsf{Salsa}\left(\begin{bmatrix} x_{3,3} & x_{3,0} & x_{3,1} & x_{3,2} \\ x_{0,3} & x_{0,0} & x_{0,1} & x_{0,2} \\ x_{1,3} & x_{1,0} & x_{1,1} & x_{1,2} \\ x_{2,3} & x_{2,0} & x_{2,1} & x_{2,2} \end{bmatrix} \right) ;$$

- the Salsa20 core operations are almost compatible with rotation of each input word by, say, 10 bits.

This shift and rotation structures are eliminated by the use of fixed constants in the input diagonal. Precisely, the input diagonal is different from all its nontrivial shifts and all its nontrivial rotations and all nontrivial shifts of its nontrivial rotations. In other words, two distinct arrays with this diagonal are always in distinct orbits under the shift/rotate group.

We are not aware of similar properties for the case of the ChaCha permutation. In particular, we are not aware of any study of the rotational properties of the ChaCha permutation.

[4] According to the authors and to [3], most of these results were already informally observed by Matt Robshaw in June 2005, and independently posted to sci.crypt by David Wagner in September 2005, but we could not find any reference besides [16].

2 ChaCha permutation description

In this section, we first define our notation, then we describe the specifications of ChaCha permutation. We do not describe the entire details of ChaCha as a stream cipher.

2.1 Notation

Let $\mathbb{F}_2$ be the binary field with two elements, and $\mathcal{M}_{\mathsf{n}\times\mathsf{n}}(\mathbb{F}_2^{\mathsf{w}})$ the set of all $\mathsf{n}\times\mathsf{n}$ matrices with elements in $\mathbb{F}_2^{\mathsf{w}}$. We indicate with lowercase letters w-bit words, i.e. $x \in \mathbb{F}_2^{\mathsf{w}}$, with bold lower case letters vectors of n words, i.e. $\boldsymbol{x} \in (\mathbb{F}_2^{\mathsf{w}})^{\mathsf{n}}$, and with uppercase letters a $\mathsf{n}\times\mathsf{n}$ matrix of n^2 words, i.e. $X \in \mathcal{M}_{\mathsf{n}\times\mathsf{n}}(\mathbb{F}_2^{\mathsf{w}})$.

We use the following notation:

- $\oplus$ for the bitwise exclusive or (XOR), i.e. the addition in $\mathbb{F}_2^{\mathsf{w}}$;
- $\boxplus$ for the w-bit addition $\mod 2^{\mathsf{w}}$;
- $\boxplus_{i=1}^{k} a_i$ for the w-bit addition $\mod 2^{\mathsf{w}}$ of k words $a_1, \ldots, a_k$
- $\lll \mathsf{r}$ and $\ggg \mathsf{r}$ for constant-distance left and right, respectively, circular rotation of r bits of a w-bit word (with $\mathsf{w} > \mathsf{r}$). When needed, we also use the following more compact notation:
  - $\overleftarrow{x}^{\mathsf{r}} = x \lll \mathsf{r}$;
  - $\overleftarrow{\boldsymbol{x}}^{\mathsf{r}} = (\overleftarrow{x_0}^{\mathsf{r}}, \ldots, \overleftarrow{x_{\mathsf{n}-1}}^{\mathsf{r}})$ the parallel left circular rotation of a n-word vector
  - $\overleftarrow{X}^{\mathsf{r}} = \begin{bmatrix} \overleftarrow{x_{0,0}}^{\mathsf{r}} & \ldots & \overleftarrow{x_{0,\mathsf{n}-1}}^{\mathsf{r}} \\ \vdots & \ddots & \vdots \\ \overleftarrow{x_{\mathsf{n}-1,0}}^{\mathsf{r}} & \ldots & \overleftarrow{x_{\mathsf{n}-1,\mathsf{n}-1}}^{\mathsf{r}} \end{bmatrix}$ the parallel left circular rotation of the w-bit elements of the matrix $X \in \mathcal{M}_{\mathsf{n}\times\mathsf{n}}(\mathbb{F}_2^{\mathsf{w}})$.

  When clear from the context, we omit the subscript r, and simply write $\overleftarrow{x}$, $\overleftarrow{\boldsymbol{x}}$, and $\overleftarrow{X}$.

In the case of ChaCha, we have $\mathsf{n} = 4$ and $\mathsf{w} = 32$.

2.2 ChaCha permutation specification

ChaCha permutation has a state of 512 bits, which can be seen as a 4×4 matrix whose elements are binary vectors of $\mathsf{w} = 32$ bits, i.e.

$$X = \{x_{i,j}\}_{\substack{i=0,\ldots,3 \\ j=0,\ldots,3}} = \begin{bmatrix} x_{0,0} & x_{0,1} & x_{0,2} & x_{0,3} \\ x_{1,0} & x_{1,1} & x_{1,2} & x_{1,3} \\ x_{2,0} & x_{2,1} & x_{2,2} & x_{2,3} \\ x_{3,0} & x_{3,1} & x_{3,2} & x_{3,3} \end{bmatrix} \in \mathcal{M}_{\mathsf{n}\times\mathsf{n}}(\mathbb{F}_2^{\mathsf{w}}) .$$

Definition 1 (ChaCha quarter round). *Let $x_i, y_i, i = 0, 1, 2, 3$ be w-bit words, and let $(y_0, y_1, y_2, y_3) = \mathcal{Q}(x_0, x_1, x_2, x_3)$, where $\mathcal{Q}$ is ChaCha quarter round, defined as follows:*

$$b_0 = x_0 \boxplus x_1 \tag{1}$$
$$b_3 = (b_0 \oplus x_3) \lll r_1 \tag{2}$$
$$b_2 = b_3 \boxplus x_2 \tag{3}$$
$$b_1 = (b_2 \oplus x_1) \lll r_2. \tag{4}$$

and

$$y_0 = b_0 \boxplus b_1 \tag{5}$$
$$y_3 = (y_0 \oplus b_3) \lll r_3 \tag{6}$$
$$y_2 = y_3 \boxplus b_2 \tag{7}$$
$$y_1 = (y_2 \oplus b_1) \lll r_4 \tag{8}$$

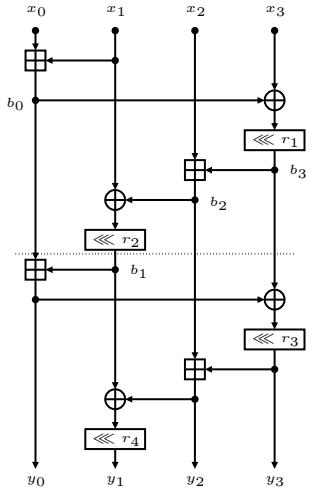

Fig. 1: The ChaCha quarter round.

We show in Fig. 1 a schematic drawing of the Chacha quarter round. The permutation used in ChaCha20 stream cipher performs 20 rounds or, equivalently, 10 *double rounds*. Two consecutive rounds (or a *double round*) of ChaCha permutation consist in applying the quarter round four times in parallel to the columns of the state (first round), and then four times in parallel to the diagonals of the state (second round). More formally:

Definition 2 (ChaCha column/diagonal round). *Let* $X = \{x_{i,j}\}_{\substack{i=0,\dots,3\\ j=0,\dots,3}}$ *and* $Y = \{y_{i,j}\}_{\substack{i=0,\dots,3\\ j=0,\dots,3}}$ *be two matrices in* $\mathcal{M}_{\mathsf{n}\times\mathsf{n}}(\mathbb{F}_2^{\mathsf{w}})$.

A column round $Y = \mathcal{R}^{\mathsf{C}}(X)$ *is defined as follows, with* $i = 0, 1, 2, 3$*:*

$$(y_{0,i}, y_{1,i}, y_{2,i}, y_{3,i}) = \mathcal{Q}(x_{0,i}, x_{1,i}, x_{2,i}, x_{3,i})\,.$$

A diagonal round $Y = \mathcal{R}^{\mathsf{D}}(X)$ *is defined as follows, for* $i = 0, 1, 2, 3$ *and where each pedix is computed modulo* $\mathsf{n} = 4$*:*

$$(y_{0,i}, y_{1,i+1}, y_{2,i+2}, y_{3,i+3}) = \mathcal{Q}(x_{0,i}, x_{1,i+1}, x_{2,i+2}, x_{3,i+3})\,.$$

3 Propagation of rotational pairs

In this section, we first define a set of necessary and sufficient conditions for the propagation of rotational pairs through ChaCha quarter round. We then use these conditions to derive a lower and an upper bound for the probability of this propagation to happen through the quarter round. Then, we describe how to extend the bounds to a full round, and finally to the full permutation.

3.1 Conditions for rotational pairs propagation

We are interested in studying the probability of the propagation through the quarter rounds of *rotational* pairs, i.e., of

$$\mathsf{p} = \Pr[(\overleftarrow{y_0}^{\mathsf{r}}, \overleftarrow{y_1}^{\mathsf{r}}, \overleftarrow{y_2}^{\mathsf{r}}, \overleftarrow{y_3}^{\mathsf{r}}) = \mathcal{Q}(\overleftarrow{x_0}^{\mathsf{r}}, \overleftarrow{x_1}^{\mathsf{r}}, \overleftarrow{x_2}^{\mathsf{r}}, \overleftarrow{x_3}^{\mathsf{r}})]\,. \tag{9}$$

To do so, we first prove the following proposition.

Proposition 1. *Given ChaCha quarter round* $\mathcal{Q}$ *defined as above with the non negative integers* $r_1, r_2, r_3, r_4 \leq \mathsf{w}-1$*, and given the rotational amount* $\mathsf{r} \leq \mathsf{w}-1$*, then*

$$\begin{gathered}(y_0 \lll \mathsf{r}, y_1 \lll \mathsf{r}, y_2 \lll \mathsf{r}, y_3 \lll \mathsf{r}) = \mathcal{Q}(x_0 \lll \mathsf{r}, x_1 \lll \mathsf{r}, x_2 \lll \mathsf{r}, x_3 \lll \mathsf{r}) \\ \iff \\ (x_0 \lll \mathsf{r}) \boxplus (x_1 \lll \mathsf{r}) = (x_0 \boxplus x_1) \lll \mathsf{r} \\ (b_3 \lll \mathsf{r}) \boxplus (x_2 \lll \mathsf{r}) = (b_3 \boxplus x_2) \lll \mathsf{r} \\ ((x_0 \boxplus x_1) \lll \mathsf{r}) \boxplus (b_1 \lll \mathsf{r}) = (x_0 \boxplus x_1 \boxplus b_1) \lll \mathsf{r} \\ (y_3 \lll \mathsf{r}) \boxplus ((b_3 \boxplus x_2) \lll \mathsf{r}) = (y_3 \boxplus b_3 \boxplus x_2) \lll \mathsf{r}\,.\end{gathered}$$

Proof. Let us consider what happens to the output if, instead of the input (x_0, x_1, x_2, x_3), we use the input $(x_0 \lll \mathsf{r}, x_1 \lll \mathsf{r}, x_2 \lll \mathsf{r}, x_3 \lll \mathsf{r})$, where every string is rotated r places to the left. First of all we find

$$\widetilde{b_0} = (x_0 \lll \mathsf{r}) \boxplus (x_1 \lll \mathsf{r}) \tag{10}$$

$$\widetilde{b_3} = \left(\widetilde{b_0} \oplus (x_3 \lll \mathsf{r})\right) \lll r_1 \tag{11}$$

$$\widetilde{b_2} = \widetilde{b_3} \boxplus (x_2 \lll \mathsf{r}) \tag{12}$$

$$\widetilde{b_1} = \left(\widetilde{b_2} \oplus (x_1 \lll \mathsf{r})\right) \lll r_2 \tag{13}$$

and

$$\widetilde{y_0} = \widetilde{b_0} \boxplus \widetilde{b_1} \tag{14}$$

$$\widetilde{y_3} = \left(\widetilde{y_0} \oplus \widetilde{b_3}\right) \lll r_3 \tag{15}$$

$$\widetilde{y_2} = \widetilde{y_3} \boxplus \widetilde{b_2} \tag{16}$$

$$\widetilde{y_1} = \left(\widetilde{y_2} \oplus \widetilde{b_1}\right) \lll r_4 \tag{17}$$

Now, the conditions that must be simultaneously fulfilled in order to obtain

$$(y_0 \lll \mathsf{r}, y_1 \lll \mathsf{r}, y_2 \lll \mathsf{r}, y_3 \lll \mathsf{r}) = (\widetilde{y_0}, \widetilde{y_1}, \widetilde{y_2}, \widetilde{y_3}) \,, \tag{18}$$

are the following:

$$\widetilde{y_0} = y_0 \lll \mathsf{r} \iff \widetilde{b_0} \boxplus \widetilde{b_1} = (b_0 \boxplus b_1) \lll \mathsf{r} \tag{19}$$

$$\widetilde{y_3} = y_3 \lll \mathsf{r} \iff \left(\widetilde{y_0} \oplus \widetilde{b_3}\right) \lll r_3 = ((y_0 \oplus b_3) \lll r_3) \lll \mathsf{r} \tag{20}$$

$$\widetilde{y_2} = y_2 \lll \mathsf{r} \iff \widetilde{y_3} \boxplus \widetilde{b_2} = (y_3 \boxplus b_2) \lll \mathsf{r} \tag{21}$$

$$\widetilde{y_1} = y_1 \lll \mathsf{r} \iff \left(\widetilde{y_2} \oplus \widetilde{b_1}\right) \lll r_4 = ((y_2 \oplus b_1) \lll r_4) \lll \mathsf{r} \tag{22}$$

These constraints can be simplified, observing that from (20), considering the condition $\widetilde{y_0} = y_0 \lll \mathsf{r}$ and thanks to the distributive property of bit rotation with respect to $\oplus$, we have

$$\begin{aligned}\left((y_0 \lll \mathsf{r}) \oplus \widetilde{b_3}\right) \lll r_3 &= ((y_0 \oplus b_3) \lll r_3) \lll \mathsf{r} \\ &= ((y_0 \lll \mathsf{r}) \oplus (b_3 \lll \mathsf{r})) \lll r_3 \,.\end{aligned}$$

Thus, we must have

$$\widetilde{b_3} = b_3 \lll \mathsf{r} \tag{23}$$

and from (22) in an analogous way, using the condition $\widetilde{y_2} = y_2 \lll \mathsf{r}$, we find that

$$\widetilde{b_1} = b_1 \lll \mathsf{r} \tag{24}$$

must hold. Now considering (23) and equalities (2) and (11) we easily observe that

$$\left(\widetilde{b_0} \lll r_1\right) \oplus ((x_3 \lll \mathsf{r}) \ll r_1) = ((b_0 \lll \mathsf{r}) \lll r_1) \oplus ((x_3 \lll \mathsf{r}) \lll r_1)$$

and we find

$$\widetilde{b_0} = b_0 \lll \mathsf{r}. \tag{25}$$

In a similar way, considering (24) and equalities (4) and (13) we have

$$\left(\widetilde{b_2} \lll r_2\right) \oplus ((x_1 \lll \mathsf{r}) \ll r_2) = ((b_2 \lll \mathsf{r}) \lll r_2) \oplus ((x_1 \lll \mathsf{r}) \lll r_2)$$

obtaining

$$\widetilde{b_2} = b_2 \lll \mathsf{r}. \tag{26}$$

Thus condition (18) corresponds to the following four conditions

$$\widetilde{b_0} = b_0 \lll \mathsf{r} \tag{27}$$

$$\widetilde{b_2} = b_2 \lll \mathsf{r} \tag{28}$$

$$(b_0 \lll \mathsf{r}) \boxplus (b_1 \lll \mathsf{r}) = (b_0 \boxplus b_1) \lll \mathsf{r} \tag{29}$$

$$(y_3 \lll \mathsf{r}) \boxplus (b_2 \lll \mathsf{r}) = (y_3 \boxplus b_2) \lll \mathsf{r} \tag{30}$$

or equivalently

$$(x_0 \lll \mathsf{r}) \boxplus (x_1 \lll \mathsf{r}) = (x_0 \boxplus x_1) \lll \mathsf{r} \tag{31}$$

$$(b_3 \lll \mathsf{r}) \boxplus (x_2 \lll \mathsf{r}) = (b_3 \boxplus x_2) \lll \mathsf{r} \tag{32}$$

$$((x_0 \boxplus x_1) \lll \mathsf{r}) \boxplus (b_1 \lll \mathsf{r}) = (x_0 \boxplus x_1 \boxplus b_1) \lll \mathsf{r} \tag{33}$$

$$(y_3 \lll \mathsf{r}) \boxplus ((b_3 \boxplus x_2) \lll \mathsf{r}) = (y_3 \boxplus b_3 \boxplus x_2) \lll \mathsf{r} \tag{34}$$

□

Remark 1. Before trying to estimate the probability $\mathfrak{p}$, i.e., that all conditions (31), (32), (33) and (34) simultaneously hold, we observe that the rotation r_4 used in ChaCha quarter round function is *not* involved in *any* of these equations, neither implicitly nor explicitly.

3.2 Bounds for the quarter round

We recall the result obtained in Corollary 4.12 by Daum [12] on the propagation of the rotational probability with respect to modular addition.

Proposition 2. *Let a and b be independent and uniformly distributed strings of w bits, and $1 \le \mathsf{r} \le \mathsf{w}-1$ an integer. Then*

$$\begin{aligned} D &= \Pr[(a \lll \mathsf{r}) \boxplus (b \lll \mathsf{r}) = (a \boxplus b) \lll \mathsf{r}] = \\ &= \frac{1 + 2^{-(\mathsf{w}-\mathsf{r})} + 2^{-\mathsf{r}} + 2^{-\mathsf{w}}}{4} = \frac{(2^{\mathsf{r}}+1)(2^{\mathsf{w}-\mathsf{r}}+1)}{2^{\mathsf{w}+2}}. \end{aligned} \tag{35}$$

The previous result can be generalized for the case where we have more than 2 addends.

Proposition 3. *Let $a_1, a_2, \ldots, a_k$ be independent and uniformly distributed strings of w bits, and $1 \le \mathsf{r} \le \mathsf{w}-1$ an integer. Then*

$$\Pr\left[\boxplus_{i=1}^{k} (a_i \lll \mathsf{r}) = \left(\boxplus_{i=1}^{k} a_i\right) \lll \mathsf{r}\right] = \frac{F(\mathsf{r},k,\mathsf{w})F(\mathsf{w}-\mathsf{r},k,\mathsf{w})}{2^{k\mathsf{w}}} \tag{36}$$

where, for $1 \le q \le \mathsf{w}-1$,

$$F(q,k,\mathsf{w}) = \sum_{h=0}^{\left\lfloor \frac{k(2^q-1)}{2^{\mathsf{w}}} \right\rfloor} \sum_{j=0}^{k} (-1)^j \binom{k}{j} \left(\binom{h2^{\mathsf{w}} - (j-1)2^q - 1 + k}{h2^{\mathsf{w}} - (j-1)2^q - 1} - \binom{h2^{\mathsf{w}} - j2^q - 1 + k}{h2^{\mathsf{w}} - j2^q - 1} \right). \tag{37}$$

Proof. In order to evaluate the number of solutions to

$$\boxplus_{i=1}^{k} (a_i \lll \mathsf{r}) = \left(\boxplus_{i=1}^{k} a_i\right) \lll \mathsf{r} \tag{38}$$

i.e., how many w-bit words $a_1, \ldots, a_k$ satisfy (38), we represent every binary string a_i and its left rotation by r as integers:

$$a_i = a_i^L 2^{\mathsf{w}-\mathsf{r}} + a_i^R,$$
$$a_i \lll \mathsf{r} = a_i^R 2^{\mathsf{r}} + a_i^L$$

where $0 \leq a_i^L \leq 2^{\mathsf{r}} - 1$ and $0 \leq a_i^R \leq 2^{\mathsf{w}-\mathsf{r}} - 1$, are, respectively, the integers represented by the left r places and right $\mathsf{w} - \mathsf{r}$ places of the binary string a_i. Now, if

$$\sum_{i=1}^{k} a_i^L = m2^{\mathsf{r}} + w, \quad \sum_{i=1}^{k} a_i^R = t2^{\mathsf{w}-\mathsf{r}} + s \tag{39}$$

with m, t, w, s non negative integers such that $w \leq 2^{\mathsf{r}} - 1$ and $s \leq 2^{\mathsf{w}-\mathsf{r}} - 1$, we have

$$\left(\boxplus_{i=1}^{k} a_i\right) = \left(\left(\sum_{i=1}^{k} a_i^L\right) 2^{\mathsf{w}-\mathsf{r}} + \sum_{i=1}^{k} a_i^R\right) \mod 2^{\mathsf{w}} = ((w+t)2^{\mathsf{w}-\mathsf{r}} + s) \mod 2^{\mathsf{w}}$$

thus, since $s \leq 2^{\mathsf{w}-\mathsf{r}} - 1$

$$\left(\boxplus_{i=1}^{k} a_i\right) \lll \mathsf{r} = s2^{\mathsf{r}} + u, \quad u = (w+t) \mod 2^{\mathsf{r}} \tag{40}$$

On the other hand

$$\boxplus_{i=1}^{k} (a_i \lll \mathsf{r}) = \left(\left(\sum_{i=1}^{k} a_i^R\right) 2^{\mathsf{r}} + \sum_{i=1}^{k} a_i^L\right) \mod 2^{\mathsf{w}} = ((s+m)2^{\mathsf{r}} + w) \mod 2^{\mathsf{w}}$$

and since $w \leq 2^{\mathsf{r}} - 1$

$$\boxplus_{i=1}^{k} (a_i \lll \mathsf{r}) = v2^{\mathsf{r}} + w, \quad v = (s+m) \mod 2^{\mathsf{w}-\mathsf{r}} \tag{41}$$

Thus from (40) and (41) we have that (38) holds if and only if

$$s2^{\mathsf{r}} + u = v2^{\mathsf{r}} + w \tag{42}$$

Hence by (40) and since $w \leq 2^{\mathsf{r}} - 1$ we have

$$u = (w+t) \bmod 2^r = w \bmod 2^r$$

which implies

$$t = c2^{\mathsf{r}} \quad c \in \mathbb{N} \tag{43}$$

and also that $u = w$. Thus we find from equality (42) that s must be equal to v, which implies

$$m = d2^{\mathsf{w}-\mathsf{r}} \quad d \in \mathbb{N} \tag{44}$$

since from (41) we have $v = (s+m) \bmod 2^{\mathsf{w}-\mathsf{r}}$. Therefore substituting (43) and (44) in (39) we observe that in order to count the number of solutions to (38) we have to count the number $F(q,k,\mathsf{w})$ of solutions in non negative integers y_i of the systems

$$\begin{cases} \sum_{i=1}^{k} y_i = h2^{\mathsf{w}} + l \\ 0 \leq y_i \leq 2^q - 1 \end{cases} \tag{45}$$

with $l = 0, 1, \ldots, 2^q - 1$ and where $h = 0, \ldots, \left\lfloor \frac{k(2^q-1)}{2^{\mathsf{w}}} \right\rfloor$ since we must have $h2^{\mathsf{w}} + l \leq k(2^q - 1)$ or equivalently

$$h + \frac{l}{2^{\mathsf{w}}} \leq \frac{k(2^q-1)}{2^{\mathsf{w}}}, \quad 0 \leq \frac{l}{2^{\mathsf{w}}} < 1$$

Thanks to Theorem 4.3 p. 138 of [11] the number of solutions of (45) for a fixed value of l is

$$\sum_{j=0}^{k} (-1)^j \binom{k}{j} \binom{h2^{\mathsf{w}} + l - j2^q + k - 1}{h2^{\mathsf{w}} + l - j2^q},$$

thus summing for all the values of l gives

$$\sum_{l=0}^{2^q-1} \sum_{j=0}^{k} (-1)^j \binom{k}{j} \binom{h2^{\mathsf{w}} + l - j2^q + k - 1}{h2^{\mathsf{w}} + l - j2^q} =$$

$$= \sum_{j=0}^{k} (-1)^j \binom{k}{j} \sum_{i=h2^{\mathsf{w}} - j2^q}^{h2^{\mathsf{w}} - (j-1)2^q - 1} \binom{i+k-1}{i} =$$

$$= \sum_{j=0}^{k} (-1)^j \binom{k}{j} \left(\binom{h2^{\mathsf{w}} - (j-1)2^q - 1 + k}{h2^{\mathsf{w}} - (j-1)2^q - 1} - \binom{h2^{\mathsf{w}} - j2^q - 1 + k}{h2^{\mathsf{w}} - j2^q - 1} \right)$$

and with a final summation on the values of h we obtain (37). Therefore the number of solutions to (38) clearly is the product $F(\mathsf{r},k,\mathsf{w})F(\mathsf{w}-\mathsf{r},k,\mathsf{w})$ and since we have $2^{k\mathsf{w}}$ possible choices for the k w-bit strings a_i we easily obtain (36). □

Remark 2. In Proposition 3 we have derived a formula for the probability that equality (38) holds. This result is in general different from the one on chained modular addictions in Lemma 2 of [14] since we do not deal with a chain and so we do not request that all the conditions similar to (38) involving $a_1, a_2, \ldots, a_h$ with $h = 2, \ldots, k-1$ must also be simultaneously satisfied.

Corollary 1. *Let a, b, c be independent and uniformly distributed strings of w bits, and $1 \leq \mathsf{r} \leq \mathsf{w} - 1$ an integer. Then*

$$\begin{aligned} &\Pr\left[(a \lll \mathsf{r}) \boxplus (b \lll \mathsf{r}) \boxplus (c \lll \mathsf{r}) = (a \boxplus b \boxplus c) \lll \mathsf{r}\right] = \\ &= \frac{D(2^{\mathsf{r}}+2)(2^{\mathsf{w}-\mathsf{r}}+2)}{9 \cdot 2^{\mathsf{w}}} + \mathbb{1}_{\{\mathsf{r}=1 \vee \mathsf{r}=\mathsf{w}-1\}} \frac{4}{2^{3\mathsf{w}}} \binom{2^{\mathsf{w}-1}}{2^{\mathsf{w}-1}-3} = P(\mathsf{r},\mathsf{w}) \end{aligned} \tag{46}$$

where with $\mathbb{1}_Z$ we indicate the usual characteristic function of Z, which is equal to 1 when Z is true and equal to 0 when Z is false.

Proof. If we use formula (36) with $k = 3$ and $1 \leq q \leq \mathsf{w} - 1$ we observe that

$$\frac{3(2^q - 1)}{2^{\mathsf{w}}} = \frac{2^{q+1} + 2^q - 3}{2^{\mathsf{w}}} = \frac{2^{q+1}}{2^{\mathsf{w}}} + \frac{2^q - 3}{2^{\mathsf{w}}}$$

thus since $0 < \frac{2^q - 3}{2^{\mathsf{w}}} < 1$ we have

$$\left\lfloor \frac{3\,(2^q - 1)}{2^{\mathsf{w}}} \right\rfloor = 1$$

when $q = \mathsf{w} - 1$. Therefore when we use (36) with $\mathsf{r} = \mathsf{w} - 1$ or, equivalently, $\mathsf{r} = 1$ we find from (37)

$$F(1, 3, \mathsf{w}) = 4, \quad F(\mathsf{w} - 1, 3, \mathsf{w}) = \binom{2^{\mathsf{w}-1} + 2}{2^{\mathsf{w}-1} - 1} + \binom{2^{\mathsf{w}-1}}{2^{\mathsf{w}-1} - 3}$$

while if $2 \leq \mathsf{r} \leq \mathsf{w} - 2$ we have

$$F(\mathsf{r}, 3, \mathsf{w}) = \binom{2^{\mathsf{r}} + 2}{2^{\mathsf{r}} - 1}, \quad F(\mathsf{w} - \mathsf{r}, 3, \mathsf{w}) = \binom{2^{\mathsf{w}-\mathsf{r}} + 2}{2^{\mathsf{w}-\mathsf{r}} - 1}$$

since in these situations $0 < \frac{3(2^q - 1)}{2^{\mathsf{w}}} < 1$ for $q = \mathsf{r}, \mathsf{w} - \mathsf{r}$. Thus a straightforward calculation shows that (46) holds. □

Remark 3. We observe that when $2 \leq \mathsf{r} \leq \mathsf{w} - 2$ this result shows a value equal to case $k = 3$ of Lemma 2 in [14] , in which we have the probability of chained modular additions for $a_1, a_2, \ldots, a_k$, w-bits words chosen at random given by

$$\Pr[\mathcal{E}] = \frac{1}{2^{3\mathsf{w}}} \binom{2^{\mathsf{r}} + 2}{2^{\mathsf{r}} - 1} \binom{2^{\mathsf{w}-\mathsf{r}} + 2}{2^{\mathsf{w}-\mathsf{r}} - 1} = \frac{D(2^{\mathsf{r}} + 2)(2^{\mathsf{w}-\mathsf{r}} + 2)}{9 \cdot 2^{\mathsf{w}}} \tag{47}$$

where

$$\begin{aligned} \mathcal{E} = &[(a_1 \boxplus a_2) \lll \mathsf{r} = (a_1 \lll \mathsf{r}) \boxplus (a_2 \lll \mathsf{r})] \cap \\ &[(a_1 \boxplus a_2 \boxplus a_3) \lll \mathsf{r} = (a_1 \lll \mathsf{r}) \boxplus (a_2 \lll \mathsf{r}) \boxplus (a_3 \lll \mathsf{r})] \,. \end{aligned} \tag{48}$$

On the other hand when $\mathsf{r} = 1$ or $\mathsf{r} = \mathsf{w} - 1$ from (46) we find the different value

$$P(1, \mathsf{w}) = P(\mathsf{w} - 1, \mathsf{w}) = \frac{4(2^{2\mathsf{w}-3} + 1)}{3 \cdot 2^{2\mathsf{w}}} \,.$$

which is greater than the corresponding one given by (47) This is an immediate consequence of the fact that if $k = 3$ we have one more addend to be considered in (37) only when $\mathsf{r} = 1, \mathsf{w} - 1$, i. e., more solutions to (38) than to the system of equalities in (48).

We now show how to obtain an upper and lower bound for the rotational probability.

Theorem 1. *The rotational probability* $\mathfrak{p}$ *of a single ChaCha quarter round is such that,*

$$D^3 P(\mathsf{r},\mathsf{w}) \leq \mathfrak{p} \leq \left(\frac{D(2^{\mathsf{r}}+2)(2^{\mathsf{w}-\mathsf{r}}+2)}{9\cdot 2^{\mathsf{w}}}\right)^2 \tag{49}$$

Proof. Let us suppose that we can couple equations (31), (33) and equations (32), (34), considering respectively x_0, x_1, b_1 and y_3, b_3, x_2 as two triplets of random w-bits words. Then we may find an upper bound for $\mathfrak{p}$ multiplying the probabilities of the two chains

$$\mathcal{E}_1 = [(x_0 \boxplus x_1) \lll \mathsf{r} = (x_0 \lll \mathsf{r}) \boxplus (x_1 \lll \mathsf{r})] \cap$$

$$[(x_0 \boxplus x_1 \boxplus b_1) \lll \mathsf{r} = (x_0 \lll \mathsf{r}) \boxplus (x_1 \lll \mathsf{r}) \boxplus (b_1 \lll \mathsf{r})]$$

$$\mathcal{E}_2 = [(b_3 \boxplus x_2) \lll \mathsf{r} = (b_3 \lll \mathsf{r}) \boxplus (x_2 \lll \mathsf{r})] \cap$$

$$[(y_3 \boxplus b_3 \boxplus x_2) \lll \mathsf{r} = (y_3 \lll \mathsf{r}) \boxplus (b_3 \lll \mathsf{r}) \boxplus (x_2 \lll \mathsf{r})]$$

obtaining $\mathfrak{p} \leq \Pr[\mathcal{E}_1]\Pr[\mathcal{E}_2] = \Pr[\mathcal{E}]^2$, i.e., $\mathfrak{p} \leq \left(\frac{D(2^{\mathsf{r}}+2)(2^{\mathsf{w}-\mathsf{r}}+2)}{9\cdot 2^{\mathsf{w}}}\right)^2$ since in the real situation, where those triplets of words are in general not all independent, there are less possible values which satisfy conditions (31), (32), (33), (34), with respect to all possible value that they may assume.

In order to obtain a lower bound we observe that (31) and (32) hold with probability D since we may consider (x_0, x_1) and (b_3, x_2) as couples of independent and uniformly distributed random variables. Moreover we may also request the restrictive condition that $(y_3, b_3 \boxplus x_2)$ are independent and uniformly distributed random variables such that also (34) hold with probability D, considering for (33) the probability given by (46) and obtaining $\mathfrak{p} \geq D^3 P(\mathsf{r},\mathsf{w})$. Thus we have $D^3 P(\mathsf{r},\mathsf{w}) \leq \mathfrak{p} \leq \left(\frac{D(2^{\mathsf{r}}+2)(2^{\mathsf{w}-\mathsf{r}}+2)}{9\cdot 2^{\mathsf{w}}}\right)^2$. □

3.3 Experimental result

To have an additional experimental confirmation of the correctness of the bounds in Theorem 1, we implemented a toy version of ChaCha quarter round, using smaller word bit size and several different combinations of round rotations $r_0, r_1, r_2, r_3,$. To run the experiment, we exhaustively search through all possible values of (x_0, x_1, x_2, x_3), then we computed $(\overleftarrow{x_0}^{\mathsf{r}}, \ldots, \overleftarrow{x_3}^{\mathsf{r}})$, evaluated both 4tuples over the quarter round function $\mathcal{Q}$, and finally checked if the condition $(\overleftarrow{y_0}^{\mathsf{r}}, \ldots, \overleftarrow{y_3}^{\mathsf{r}}) = \mathcal{Q}(\overleftarrow{x_0}^{\mathsf{r}}, \ldots, \overleftarrow{x_3}^{\mathsf{r}})$ was verified, and counted how many times we would happen (#collisions column in Table 1). In Table 1, we show some of the results for word size of 4, 5, and 6 bits. The value p is the probability to have a rotational collision for a random permutation f, i.e. $p = \Pr[(\overleftarrow{y_0}^{\mathsf{r}}, \ldots, \overleftarrow{y_3}^{\mathsf{r}}) = f(\overleftarrow{x_0}^{\mathsf{r}}, \ldots, \overleftarrow{x_3}^{\mathsf{r}})]$. Notice that the case r is equal to $\mathsf{w} - \mathsf{r}$, so we do not report it in the table.

$\mathsf{w} = 4, (r_0, r_1, r_2, r_3) = (1, 3, 2, 1)$					
r	#collisions	Lower Bound	$\mathfrak{p}$	Upper Bound	p
1	747	$0.00880 \sim 2^{-6.83}$	**0.01140**	$0.01373 \sim 2^{-6.19}$	$2^{-16.00}$
2	388	$0.00582 \sim 2^{-7.42}$	**0.00592**	$0.00954 \sim 2^{-6.71}$	$2^{-16.00}$

$\mathsf{w} = 5, (r_1, r_2, r_3, r_4) = (4, 3, 2, 1)$					
r	#collisions	Lower Bound	$\mathfrak{p}$	Upper Bound	p
1	8917	$0.00630 \sim 2^{-7.31}$	**0.00850**	$0.00992 \sim 2^{-6.66}$	$2^{-20.00}$
2	3405	$0.00318 \sim 2^{-8.30}$	**0.00325**	$0.00536 \sim 2^{-7.54}$	$2^{-20.00}$

$\mathsf{w} = 6, (r_1, r_2, r_3, r_4) = (5, 3, 2, 1)$					
r	#collisions	Lower Bound	$\mathfrak{p}$	Upper Bound	p
1	123317	$0.00528 \sim 2^{-7.57}$	**0.00735**	$0.00834 \sim 2^{-6.91}$	$2^{-24.00}$
2	39482	$0.00228 \sim 2^{-8.78}$	**0.00235**	$0.00388 \sim 2^{-8.01}$	$2^{-24.00}$
3	32628	$0.00174 \sim 2^{-9.17}$	**0.00194**	$0.00302 \sim 2^{-8.37}$	$2^{-24.00}$

Table 1. Experimental results on a toy version of ChaCha quarter round.

3.4 Bounds propagation through the full round

We indicate with

- $Y = \mathcal{R}(X)$ the application of one round of the ChaCha permutation (either a column or a diagonal round).
- $Y = \mathcal{R}^i(X)$ the application of i consecutive round of the ChaCha permutation, alternating column to diagonal rounds (where the first round $\mathcal{R}^1$ is a column round).

The following theorem shows how to extend the lower and upper bounds of Theorem 1 from the ChaCha quarter round to one full round of the ChaCha permutation.

Theorem 2. *Let* L, U *be such that* $\mathsf{L} \leq \Pr\left[\overleftarrow{\boldsymbol{y}} = \mathcal{Q}(\overleftarrow{\boldsymbol{x}})\right] \leq \mathsf{U}$. *Then*

$$\mathsf{L}^{\mathsf{n}} \leq \Pr\left[\overleftarrow{Y} = \mathcal{R}(\overleftarrow{X})\right] \leq \mathsf{U}^{\mathsf{n}} \tag{50}$$

Proof. Since a full round applies n quarter rounds independently in parallel, to extend the bounds from Theorem 1 it is sufficient to multiply the probabilities, i.e., for the rounds where the quarter round is applied to the columns we have

$$\Pr\left[\overleftarrow{Y} = \mathcal{R}(\overleftarrow{X})\right] =$$

$$\Pr\left[\begin{pmatrix}\overleftarrow{y_{0,0}}\\ \vdots \\ \overleftarrow{y_{\mathsf{n}-1,0}}\end{pmatrix} = \mathcal{Q}\begin{pmatrix}\overleftarrow{x_{0,0}}\\ \vdots \\ \overleftarrow{y_{\mathsf{n}-1,0}}\end{pmatrix} \wedge \cdots \wedge \begin{pmatrix}\overleftarrow{y_{0,\mathsf{n}-1}}\\ \vdots \\ \overleftarrow{y_{\mathsf{n}-1,\mathsf{n}-1}}\end{pmatrix} = \mathcal{Q}\begin{pmatrix}\overleftarrow{x_{0,\mathsf{n}-1}}\\ \vdots \\ \overleftarrow{y_{\mathsf{n}-1,\mathsf{n}-1}}\end{pmatrix}\right] =$$

$$\Pr\left[\begin{pmatrix}\overleftarrow{y_{0,0}}\\ \vdots \\ \overleftarrow{y_{\mathsf{n}-1,0}}\end{pmatrix} = \mathcal{Q}\begin{pmatrix}\overleftarrow{x_{0,0}}\\ \vdots \\ \overleftarrow{y_{\mathsf{n}-1,0}}\end{pmatrix}\right] \cdot \ldots \cdot \Pr\left[\begin{pmatrix}\overleftarrow{y_{0,\mathsf{n}-1}}\\ \vdots \\ \overleftarrow{y_{\mathsf{n}-1,\mathsf{n}-1}}\end{pmatrix} = \mathcal{Q}\begin{pmatrix}\overleftarrow{x_{0,\mathsf{n}-1}}\\ \vdots \\ \overleftarrow{y_{\mathsf{n}-1,\mathsf{n}-1}}\end{pmatrix}\right].$$

For the rounds where the quarter round is applied to the diagonals, the proof is alike. □

Recall that, in Theorem 1, for $\mathsf{n} = 4$, we proved that $\mathsf{L} = D^3 P(\mathsf{r}, \mathsf{w})$ and $\mathsf{U} = \left(\frac{D(2^{\mathsf{r}}+2)(2^{\mathsf{w}-\mathsf{r}}+2)}{9 \cdot 2^{\mathsf{w}}}\right)^2$.

3.5 Bounds propagation through the full permutation

The following theorem shows how to extend the lower and upper bounds of Theorem 2 from one round of ChaCha to i consecutive rounds. To prove the theorem, we make an assumption that seems to be a good approximation of what happens in practice, i.e. we assume that the input states of each round are independent and uniformly distributed.

Theorem 3. *Let* L, U *be such that* $\mathsf{L} \leq \Pr\left[\overleftarrow{\boldsymbol{y}} = \mathcal{Q}(\overleftarrow{\boldsymbol{x}})\right] \leq \mathsf{U}$. *Then*

$$\mathsf{L}^{\mathsf{n}i} \leq \Pr\left[\overleftarrow{Y} = \mathcal{R}^i(\overleftarrow{X})\right] \leq \mathsf{U}^{\mathsf{n}i} \tag{51}$$

Proof. Because of Theorem 2, we have that $\mathsf{L}^{\mathsf{n}} \leq \Pr\left[\overleftarrow{Y} = \mathcal{R}^1(\overleftarrow{X})\right] \leq \mathsf{U}^{\mathsf{n}}$. For the inductive step, notice that $\Pr\left[\overleftarrow{Y} = \mathcal{R}^i(\overleftarrow{X})\right] = \Pr\left[\overleftarrow{Y} = \mathcal{R}(\mathcal{R}^{i-1}(\overleftarrow{X}))\right]$. Thus, for the assumption of independence of each state, we have that the equality $\Pr\left[\overleftarrow{Y} = \mathcal{R}(\mathcal{R}^{i-1}(\overleftarrow{X}))\right] = \Pr\left[\overleftarrow{Y} = \mathcal{R}(\overleftarrow{\mathcal{R}^{i-1}(X)})\right]$ holds with probability bounded by L^{n} and U^{n}. □

4 Distinguisher description

When $\overleftarrow{\boldsymbol{y}}^{\mathsf{r}} = F(\overleftarrow{\boldsymbol{x}}^{\mathsf{r}})$, we say that F has a *parallel rotational collision* (or simply a *rotational collision*) in $\boldsymbol{x}$ with respect to r. In this section, we show that, up to a certain number of rounds, ChaCha permutation has more rotational collisions with respect to a random permutation with a fixed point. We first describe what is the probability to have a rotational collision for a random permutation Π with a fixed point. Then, we use this probability and the bounds from subsection 3.5 to distinguish ChaCha permutation from Π.

4.1 Rotational collisions of a random permutation

For every set A, let $\mathcal{S}(A)$ be the group of permutations over A. Moreover, for each permutation $\Pi : (\mathbb{F}_2^{\mathsf{w}})^k \to (\mathbb{F}_2^{\mathsf{w}})^k$ let $C_\Pi := \#\left\{\boldsymbol{x} \in (\mathbb{F}_2^{\mathsf{w}})^k : \Pi(\overleftarrow{\boldsymbol{x}}) = \overleftarrow{\Pi(\boldsymbol{x})}\right\}$ be the number of rotational collisions of Π. We want first to compute the expected number of rotational collisions of a random permutation.

Proposition 4. *We have* $\#\{\boldsymbol{x} \in \left(\mathbb{F}_2^{\mathsf{w}}\right)^k : \boldsymbol{x} = \overleftarrow{\boldsymbol{x}}\} = 2^{k \gcd(\mathsf{w},\mathsf{r})}$.

Proof. For each $\boldsymbol{x} = (x_1, \ldots, x_k) \in \left(\mathbb{F}_2^{\mathsf{w}}\right)^k$ we have $\boldsymbol{x} = \overleftarrow{\boldsymbol{x}}$ if and only if $x_i = \overleftarrow{x_i}$ for each $i \in \{1, \ldots, k\}$. Hence, it is enough to show that $\#\{x \in \mathbb{F}_2^{\mathsf{w}} : x = \overleftarrow{x}\} = 2^{\gcd(\mathsf{w},\mathsf{r})}$. In turn, this is equivalent to the assertion that the permutation of $\mathbb{Z}/\mathsf{w}\mathbb{Z}$ given by $k \mapsto k + \mathsf{r}$ has $\gcd(\mathsf{w}, \mathsf{r})$ cycles, which is a well-known fact. □

We can now compute the expected number of rotational collisions of a random permutation.

Proposition 5. *Let* Π *be a uniformly random variable in* $\mathcal{S}\big((\mathbb{F}_2^{\mathsf{w}})^k\big)$. *Then*

$$\mathbb{E}[C_\Pi] = \frac{2^{\mathsf{w}k} + 2^{2k\gcd(\mathsf{w},\mathsf{r})} - 2^{k\gcd(\mathsf{w},\mathsf{r})+1}}{2^{\mathsf{w}k} - 1}.$$

Proof. By the definition of expected value, we have

$$\begin{aligned}
\mathbb{E}[C_\Pi] &= \frac{1}{\#\mathcal{S}\big((\mathbb{F}_2^{\mathsf{w}})^k\big)} \sum_{\Pi \in \mathcal{S}((\mathbb{F}_2^{\mathsf{w}})^k)} \#\big\{\boldsymbol{x} \in (\mathbb{F}_2^{\mathsf{w}})^k : \Pi(\overleftarrow{\boldsymbol{x}}) = \overleftarrow{\Pi(\boldsymbol{x})}\big\} \\
&= \frac{1}{(2^{\mathsf{w}k})!} \sum_{\Pi \in \mathcal{S}((\mathbb{F}_2^{\mathsf{w}})^k)} \sum_{\boldsymbol{x} \in (\mathbb{F}_2^{\mathsf{w}})^k} \mathbb{1}\big[\Pi(\overleftarrow{\boldsymbol{x}}) = \overleftarrow{\Pi(\boldsymbol{x})}\big] \\
&= \frac{1}{(2^{\mathsf{w}k})!} \sum_{\boldsymbol{x} \in (\mathbb{F}_2^{\mathsf{w}})^k} \sum_{\Pi \in \mathcal{S}((\mathbb{F}_2^{\mathsf{w}})^k)} \mathbb{1}\big[\Pi(\overleftarrow{\boldsymbol{x}}) = \overleftarrow{\Pi(\boldsymbol{x})}\big] \\
&= \frac{1}{(2^{\mathsf{w}k})!} \sum_{\boldsymbol{x} \in (\mathbb{F}_2^{\mathsf{w}})^k} \#\big\{\Pi \in \mathcal{S}\big((\mathbb{F}_2^{\mathsf{w}})^k\big) : \Pi(\overleftarrow{\boldsymbol{x}}) = \overleftarrow{\Pi(\boldsymbol{x})}\big\} \\
&= \frac{1}{(2^{\mathsf{w}k})!} \sum_{\boldsymbol{x},\boldsymbol{y} \in (\mathbb{F}_2^{\mathsf{w}})^k} N_{\boldsymbol{x},\boldsymbol{y}},
\end{aligned}$$

where $N_{\boldsymbol{x},\boldsymbol{y}} := \#\big\{\Pi \in \mathcal{S}\big((\mathbb{F}_2^{\mathsf{w}})^k\big) : \Pi(\boldsymbol{x}) = \boldsymbol{y} \wedge \Pi(\overleftarrow{\boldsymbol{x}}) = \overleftarrow{\boldsymbol{y}}\big\}$, for every $\boldsymbol{x}, \boldsymbol{y} \in (\mathbb{F}_2^{\mathsf{w}})^k$. Hence, we have to compute $N_{\boldsymbol{x},\boldsymbol{y}}$. There are four cases:

1. If $\boldsymbol{x} = \overleftarrow{\boldsymbol{x}}$ and $\boldsymbol{y} = \overleftarrow{\boldsymbol{y}}$, then $N_{\boldsymbol{x},\boldsymbol{y}} = (2^{\mathsf{w}k} - 1)!$
2. If $\boldsymbol{x} \neq \overleftarrow{\boldsymbol{x}}$ and $\boldsymbol{y} \neq \overleftarrow{\boldsymbol{y}}$, then $N_{\boldsymbol{x},\boldsymbol{y}} = (2^{\mathsf{w}k} - 2)!$
3. If $\boldsymbol{x} \neq \overleftarrow{\boldsymbol{x}}$ and $\boldsymbol{y} = \overleftarrow{\boldsymbol{y}}$, then $N_{\boldsymbol{x},\boldsymbol{y}} = 0$
4. If $\boldsymbol{x} = \overleftarrow{\boldsymbol{x}}$ and $\boldsymbol{y} \neq \overleftarrow{\boldsymbol{y}}$, then $N_{\boldsymbol{x},\boldsymbol{y}} = 0$

Consequently, using also Proposition 4, we get the claimed formula:

$$\begin{aligned}
\mathbb{E}[C_{\Pi}] &= \frac{1}{(2^{\mathsf{w}k})!}\left(\sum_{\substack{\boldsymbol{x},\boldsymbol{y}\in(\mathbb{F}_2^{\mathsf{w}})^k \\ \boldsymbol{x}=\overleftarrow{\boldsymbol{x}},\ \boldsymbol{y}=\overleftarrow{\boldsymbol{y}}}} (2^{\mathsf{w}k}-1)! + \sum_{\substack{\boldsymbol{x},\boldsymbol{y}\in(\mathbb{F}_2^{\mathsf{w}})^k \\ \boldsymbol{x}\neq\overleftarrow{\boldsymbol{x}},\ \boldsymbol{y}\neq\overleftarrow{\boldsymbol{y}}}} (2^{\mathsf{w}k}-2)!\right) \\
&= \frac{1}{2^{\mathsf{w}k}} \sum_{\substack{\boldsymbol{x},\boldsymbol{y}\in(\mathbb{F}_2^{\mathsf{w}})^k \\ \boldsymbol{x}=\overleftarrow{\boldsymbol{x}},\ \boldsymbol{y}=\overleftarrow{\boldsymbol{y}}}} 1 + \frac{1}{2^{\mathsf{w}k}(2^{\mathsf{w}k}-1)} \sum_{\substack{\boldsymbol{x},\boldsymbol{y}\in(\mathbb{F}_2^{\mathsf{w}})^k \\ \boldsymbol{x}\neq\overleftarrow{\boldsymbol{x}},\ \boldsymbol{y}\neq\overleftarrow{\boldsymbol{y}}}} 1 \\
&= \frac{1}{2^{\mathsf{w}k}} \cdot 2^{2k\gcd(\mathsf{w},\mathsf{r})} + \frac{1}{2^{\mathsf{w}k}(2^{\mathsf{w}k}-1)} \cdot \left(2^{\mathsf{w}k} - 2^{k\gcd(\mathsf{w},k)}\right)^2 \\
&= \frac{2^{\mathsf{w}k} + 2^{2k\gcd(\mathsf{w},\mathsf{r})} - 2^{k\gcd(\mathsf{w},\mathsf{r})+1}}{2^{\mathsf{w}k}-1}.
\end{aligned}$$

□

For $\mathsf{w} = 32$, $\mathsf{n} = 4$, and $\mathsf{r} = 1$, then $\mathbb{E}[C_{\Pi}]$ is basically 1. As a consequence, for a random permutation Π with a fixed point, then $\mathbb{E}[C_{\Pi}]$ is basically 2.

4.2 ChaCha permutation vs random permutation

In Table 2, we display the lower and upper bounds of subsection 3.5, Theorem 3, for $\mathsf{w} = 32$, $\mathsf{n} = 4$, $\mathsf{r} = 1$, and rounds from 1 to 20. As we showed in subsection 4.1, for a random permutation $\Pi \in \mathcal{S}(\mathbb{F}_2^{\mathsf{nw}})$ with one fixed point, a rotational collision happens with probability very close to $2/2^{\mathsf{nw}}$. In the case of ChaCha parameters this probability is $1/2^{511}$. Thus, we can build a distinguisher $\mathcal{A}$ with access to an oracle Oracle running either ChaCha_{π} with ρ rounds or Π. Let L^{ni} and U^{ni} be, respectively, the upper and lower bound of Theorem 3, with $i = 1, \ldots, \rho$. The algorithm $\mathcal{A}$ runs as follow: generate binary strings $X_i \in \mathbb{F}_2^{\mathsf{nw}}$ for $i = 1, \ldots, \lceil 1/\mathsf{U}^{ni} \rceil$; ask the oracle the corresponding output $Y_i = \mathsf{Oracle}(X_i)$; if there exists i such that $\overleftarrow{Y_i} = \mathsf{Oracle}(\overleftarrow{X_i})$ then the algorithm says the oracle is running ChaCha_{π}. If such i does not exists, then the oracle is running Π.

The complexity of the algorithm $\mathcal{A}$ is dominated by the $2\lceil 1/\mathsf{U}^{ni} \rceil$ calls to the oracle. For example, to distinguish ChaCha_{π} with 8 rounds, $\mathcal{A}$ performs 2^{231} calls to the oracle, while for ChaCha_{π} with 17 rounds, the calls are 2^{489}. After the 17th round, $\mathcal{A}$ can not distinguish ChaCha_{π} from Π anymore.

5 Conclusion

We showed that parallel rotational collisions are more likely to happen in ChaCha underlying permutation with up to 17 rounds, than in a random permutation of the same input size. We are not aware of any theoretical study of ChaCha rotational properties, and we leave to future research finding an application of our results to the cryptanalysis of ChaCha stream cipher.

Round	Lower Bound L^{ni}	Upper Bound U^{ni}	Round	Lower Bound L^{ni}	Upper Bound U^{ni}
1	$\sim 2^{-27.32}$	$\sim 2^{-28.68}$	11	$\sim 2^{-300.52}$	$\sim 2^{-315.48}$
2	$\sim 2^{-54.64}$	$\sim 2^{-57.36}$	12	$\sim 2^{-327.84}$	$\sim 2^{-344.16}$
3	$\sim 2^{-81.96}$	$\sim 2^{-86.04}$	13	$\sim 2^{-355.16}$	$\sim 2^{-372.84}$
4	$\sim 2^{-109.28}$	$\sim 2^{-114.72}$	14	$\sim 2^{-382.48}$	$\sim 2^{-401.52}$
5	$\sim 2^{-136.60}$	$\sim 2^{-143.40}$	15	$\sim 2^{-409.80}$	$\sim 2^{-430.20}$
6	$\sim 2^{-163.92}$	$\sim 2^{-172.08}$	16	$\sim 2^{-437.12}$	$\sim 2^{-458.88}$
7	$\sim 2^{-191.24}$	$\sim 2^{-200.76}$	17	$\sim 2^{-464.45}$	$\sim 2^{-487.55}$
8	$\sim 2^{-218.56}$	$\sim 2^{-229.44}$	18	$\sim 2^{-491.77}$	$\sim 2^{-516.23}$
9	$\sim 2^{-245.88}$	$\sim 2^{-258.12}$	19	$\sim 2^{-519.09}$	$\sim 2^{-544.91}$
10	$\sim 2^{-273.20}$	$\sim 2^{-286.80}$	20	$\sim 2^{-546.41}$	$\sim 2^{-573.59}$

Table 2. Bounds propagation through ChaCha rounds with $\mathsf{w} = 32$, $\mathsf{n} = 4$, and $\mathsf{r} = 1$.

References

1. Aumasson, J.P., Neves, S., Wilcox-OHearn, Z., Winnerlein, C.: BLAKE2: simpler, smaller, fast as MD5. In: International Conference on Applied Cryptography and Network Security. pp. 119–135. Springer (2013)
2. Bernstein, D.: Salsa20 security. Tech. rep., eSTREAM Project (2005), available at: http://cr.yp.to/snuffle/security.pdf
3. Bernstein, D.J.: Response to "On the Salsa20 core function", available at: https://cr.yp.to/snuffle/reoncore-20080224.pdf
4. Bernstein, D.J.: The Salsa20 core, available at: http://cr.yp.to/salsa20.html
5. Bernstein, D.J.: Salsa20 specification. Tech. rep., eSTREAM Project, https://www.ecrypt.eu.org/stream/ (2005), available at: http://www.ecrypt.eu.org/stream/salsa20pf.html
6. Bernstein, D.J.: Salsa20 specification. eSTREAM Project algorithm description, http://www.ecrypt.eu.org/stream/salsa20pf.html (2005)
7. Bernstein, D.J.: What output size resists collisions in a xor of independent expansions? In: ECRYPT Workshop on Hash Functions (2007)
8. Bernstein, D.J.: ChaCha, a variant of Salsa20. In: Workshop Record of SASC. vol. 8, pp. 3–5 (2008)
9. Bernstein, D.J.: The Salsa20 family of stream ciphers. In: New stream cipher designs, pp. 84–97. Springer (2008)
10. Bernstein, D.J., Hopwood, D., Hülsing, A., Lange, T., Niederhagen, R., Papachristodoulou, L., Schneider, M., Schwabe, P., Wilcox-OHearn, Z.: Sphincs: practical stateless hash-based signatures. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. pp. 368–397. Springer (2015)
11. Charalambides, C.A.: Enumerative Combinatorics. Chapman & Hall (2002)
12. Daum, M.: Cryptanalysis of Hash functions of the MD4-family. Ph.D. thesis, Ruhr University Bochum (2005), http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.88.7847&rep=rep1&type=pdf
13. Hernandez-Castro, J.C., Tapiador, J.M., Quisquater, J.J.: On the salsa20 core function. In: International Workshop on Fast Software Encryption. pp. 462–469. Springer (2008)
14. Khovratovich, D., Nikolić, I., Pieprzyk, J., Sokolowski, P., Steinfeld, R.: Rotational cryptanalysis of ARX revisited. In: Fast Software Encryption. pp. 519–536. Springer (2015)

15. Nir, Y., Langley, A.: Chacha20 and poly1305 for IETF protocols. RFC **8439**, 1–46 (2018). https://doi.org/10.17487/RFC8439, https://doi.org/10.17487/RFC8439
16. Various: Google groups: sci.crypt/re-rolled salsa20 function, newsgroup on September 26th (2005). Available at: https://groups.google.com/forum/#!msg/sci.crypt/AkQnSoO40BA/o4eG96rjkgYJ

This figure "chachaquarterround.png" is available in "png" format from:

http://arxiv.org/ps/2008.13406v1