Symmetric Twin Column Parity Mixers and their Applications

Hao Lei[1,2], Raghvendra Rohit[3], Guoxiao Liu[4], Jiahui He[1,2], Mohamed Rachidi[3], Keting Jia[4,5,6], Kai Hu[1,2](✉) and Meiqin Wang[1,2,7]

[1] School of Cyber Science and Technology, Shandong University, Qingdao, Shandong, China
leihao@mail.sdu.edu.cn,hejiahui2020@mail.sdu.edu.cn,kai.hu@sdu.edu.cn,mqwang@sdu.edu.cn

[2] Key Laboratory of Cryptologic Technology and Information Security, Ministry of Education, Shandong University, Jinan, China.

[3] Cryptography Research Centre, Technology Innovation Institute, Abu Dhabi, United Arab Emirates iraghvendrarohit@gmail.com,Mohamed.Rachidi@tii.ae

[4] Institute for Network Sciences and Cyberspace, Tsinghua University, Beijing, China
lgx22@mails.tsinghua.edu.cn,ktjia@tsinghua.edu.cn

[5] BNRist, Tsinghua University, Beijing, China

[6] Zhongguancun Laboratory, Beijing, China

[7] Quan Cheng Shandong Laboratory, Jinan, China

Abstract. The circulant twin column parity mixer (TCPM) is a type of mixing layer for the round function of cryptographic permutations designed by Hirch et al. at CRYPTO 2023. It has a bitwise differential branch number of 12 and a bitwise linear branch number of 4, which makes it competitive in applications where differential security is required. Hirch et al. gave a concrete instantiation of a permutation using such a mixing layer, named Gaston, and showed the best 3-round differential and linear trails of Gaston have much higher weights than those of Ascon. In this paper, we first prove why the TCPM has linear branch number 4 and then show that Gaston's linear behavior is worse than Ascon for more than 3 rounds. Motivated by these facts, we aim to enhance the linear security of the TCPM. We show that adding a specific set of row cyclic shifts to the TCPM can make its differential and linear branch numbers both 12. Notably, by setting a special relationship between the row shift parameters of the modified TCPM, we obtain a special kind of mixlayer called the symmetric circulant twin column parity mixer. The symmetric TCPM has a unique design property that its differential and linear branch histograms are the same, which makes the parameter selection process and the security analysis convenient. Using the symmetric TCPM, we present two new 320-bit cryptographic permutations, namely (1) Gaston-S where we replace the mixing layer in Gaston with the symmetric TCPM and (2) SBD which uses a low-latency degree-4 S-box as the non-linear layer and the symmetric TCPM as the mixing layer. We evaluate the security of these permutations considering differential, linear and algebraic analysis, and then provide the performance comparison with Gaston in both hardware and software. Our results indicate that Gaston-S and SBD are competitive with Gaston in both security and performance.

Keywords: Mixing layer · Permutations · Branch number · Column parity mixer (CPM) · Gaston · Ascon

1 Introduction

In recent years, lightweight cryptography has gained significant attention from academia and industry due to the advent of resource-constrained devices such as smart cards, RFIDs and EPC tags, IoT devices, sensor networks, etc., where computing power, storage, implementation area, and energy supply are limited. The central goal of lightweight cryptography is to minimize the overall implementation cost of cryptographic primitives across various dimensions including state and key sizes, latency, throughput, power consumption, and physical footprint, while ensuring a sufficient level of security [MFW+17, Nat19]. At a high level, the implementation cost of a primitive is closely related to the type (XOR, (N)AND, (N)OR, NOT) and number of Boolean operations.

The substitution permutation network (SPN) is one of the key design paradigms of symmetric ciphers. The SPN algorithms ensure the security of a cipher by iterating a round function composed of linear and non-linear layers multiple times. The non-linear layer typically uses substitution boxes (S-boxes in short), whose main function is to provide confusion. Its strength is usually measured by algebraic degree, differential uniformity, and nonlinearity. The linear layer is responsible for the diffusion of bits and its security is often measured by the branch number.

One of the common examples of linear layers is the well-known maximum distance separable (MDS) matrices. For instance, a 4×4 MDS matrix is used in the MixColumns operation in AES. Its (differential and linear) branch number is 5. Since MDS matrices are defined over the extension fields, they have certain implementation constraints. Due to this fact, MDS matrices used in cryptographic algorithms typically have a maximum dimension of 8×8, such as the MDS matrices in Whirlpool [BR+00] and SHARK [RDP+96] which both have branch numbers of 9. Despite these limitations, significant research has been done on finding/implementating MDS matrices with low gate cost [CTG16, DL18, KLSW17, LW17, LSS+19, LS16, SKOP15, VKS22].

Unlike MDS matrices, the linear layer of some algorithms is constructed with cyclic shifts and a small number of XOR operations, such as the mixing layer of Keccak-f [BDPVA09], Xoodoo [DHVAVK18] and Ascon [DEMS21]. In all three permutations, the mixing layer has a branch number of 4 and the implementation cost is 2 binary XOR operations per bit. Keccak-f and Xoodoo have a mixing layer of the type column parity mixer (CPM) that can not be split into a number of parallel mappings but operates on the state as a whole.

At Crypto 2023, Hirch et al. [HDRM23] introduced the TCPM which is a generalization of the CPM. The TCPM has a differential branch number of 12 and a linear branch number of 4. Its computational cost is between 3 and 3.34 binary XORs per bit (depending on the dimension). Using such a mixing layer, Hirch et al. gave a permutation named Gaston. The Gaston's round function takes as few bitwise operations as Ascon and the best 3-round differential and linear trails of Gaston have much higher weights than those of Ascon.

Since the publication of Gaston, we are not aware of any third-party cryptanalysis results on it. Thus, it is an intriguing question to ask: (1) what is the security of Gaston compared to Ascon for more than 3 rounds and (2) can we improve the linear branch of Gaston without a significant change in the implementation cost ?

In this work, we aim to address the aforementioned two questions. We give new cryptanalysis results on Gaston and introduce a new linear layer called the symmetric TCPM. In what follows, we list our contributions.

1.1 Our Contributions

1. **Linear cryptanalysis of Gaston.** We analyze the TCPM from a theoretical perspective and show why its linear branch number is 4. We then investigate the linear trails of Gaston and find a 2-round iterative trail with squared correlation 2^{-48}. We also report 4, 5, and 6 rounds linear trails of Gaston with squared correlation

2^{-82}, 2^{-120} and 2^{-144} while the best known 4 and 5 rounds linear trails of Ascon have squared correlation 2^{-98} and 2^{-184}.

2. **The modified TCPM.** We add a set of row cyclic shifts to the original TCPM to improve its linear security, and called the resulting diffusion layer as the modified TCPM. We show that for a proper choice of offsets, the modified TCPM can achieve the differential branch number and linear branch number 12. The addition of extra cyclic shifts increased the computational cost to 4 XORs per bit (from $3-3.34$ XORs per bit in Gaston), but the maximum XOR depth remains unchanged. It allows to get a much higher specific branch number for the same amount of computation than MDS matrices. In Table 1, we compare various types of mixing layers.
3. **The symmetric TCPM and its parameters.** We present a special type of the modified TCPM called the symmetric TCPM whose differential branch histogram and linear differential branch histogram are the same. This notable property makes the parameter selection process and the security analysis convenient. We discuss the diffusion properties of the symmetric TCPM, and provide the concrete parameter set for a state with dimension 5×64 along with the rationale of selection of these parameters.
4. **Design, security and implementation of Gaston-S and SBD.** We present the design of two new 320-bit cryptographic permutations, namely Gaston-S and SBD which use a symmetric TCPM as their underlying mixing layer. Gaston-S employs χ mapping as the non-linear layer while SBD utilizes a newly introduced 5-bit low-latency S-box with algebraic degree 4.

   We evaluate the differential and linear trails of Gaston-S and SBD, and find that the best 3-round trails have much higher weights than those of Gaston. Moreover, we provide the algebraic degree upper bounds of them using the division property method.

   We also implement Gaston-S and SBD in three standard cell libraries (NanGate 15nm, 45nm, and TSMC 28 nm) and two software platforms (Intel Xeon x86_64 and ARMv8). We discuss the performance comparison and report that in hardware the latency of Gaston, Gaston-S, and SBD are almost the same. In case of software performance, the speeds (in clock cycles per byte) of Gaston and Gaston-S are similar, while SBD is slower than both of them.

   The source codes for hardware and software implementation, and security evaluation of Gaston-S and SBD are available at https://github.com/lhoop/STCPM.

1.2 Outline of the Paper

The rest of the paper is organized as follows. In Section 2, we recall the basic concepts of differential and linear cryptanalysis, some metrics related to the diffusion layers, and briefly discuss (twin) column parity mixers. Section 3 presents our results on the linear cryptanalysis of Gaston. In Section 4, we explain the theory of modified and symmetric TCPMs. In Section 5, we discuss the diffusion properties of symmetric TCPMs. Section 6 provides the parameter set of the symmetric TCPM for a state with dimension 5×64. In Section 7, we give the specifications, security analysis, and implementation results of Gaston-S and SBD. We conclude the paper in Section 8 with future research directions.

2 Preliminaries

In this paper, we will study iterated permutations with a round function R consisting of a linear layer that we will denote by λ and a non-linear layer that we will denote by γ. We

Table 1: A comparison of different types of mixing layers. For MDS, the dimensions in the second column represent the dimension of the matrix (defined over the finite field with d being the degree of the field defining polynomial) while for the rest it denotes the state size in bits. m and n denote the number of rows and number of columns, respectively.

Mixing layer type	Dimensions	XORs per bit	Branch number		Source
			Diff.	Linear	
MDS	$3 \times 3, GF\left(2^d\right)$	$\frac{5}{3} + \frac{1}{3d}$	4	4	[DL18]
	$4 \times 4, GF\left(2^d\right)$	$2 + \frac{3}{4d}$	5	5	[DL18]
	$8 \times 8, GF\left(2^4\right)$	6.06	9	9	[KLSW17]
	$8 \times 8, GF\left(2^8\right)$	6.125	9	9	[KLSW17]
CPM	$5 \times 5 \times w\dagger, GF(2)$	2	4	4	[BDPVA13]
	$3 \times 4 \times 32, GF(2)$	2	4	4	[DHVAVK18]
	$m \times n, GF(2)$	$2 + \frac{h-2}{m}\ddagger$	4	4	[SD18]
Ascon p_L	$5 \times 64, GF(2)$	2	4	4	[DEMS21]
TCPM	$m \times n, GF(2)$	$3 + \frac{1}{m}$	12	4	[HDRM23]
Transpose of TCPM	$m \times n, GF(2)$	$3 + \frac{1}{m}$	4	12	[HDRM23]
The modified TCPM	$m \times n, GF(2)$	4	12	12	This work

$\dagger$: w represents the number of 5×5 slices and $w \in \{8, 16, 32, 64\}$ based on the variant of Keccak-f. $\ddagger$: h is the Hamming weight of the parity-folding polynomial as defined in [SD18].

assume $R = \gamma \circ \lambda$.

We assume the round function operates on a two-dimensional state $\boldsymbol{A}$ with m rows and n columns. We use A_i $(0 \leq i < m)$ to represent the i-th row of $\boldsymbol{A}$ and $a_{i,j}$ to represent the bit in row i and column j. Sometimes, the state is also regarded as a mn-dimensional vector where we rearrange the elements of the matrix according to the row-first principle. In other words, $\boldsymbol{A}$ will be a cascade of $A_i, 0 \leq i < m$. We denote the set of all possible states by $\mathcal{A}$. Also, we assume the non-linear layer operates in parallel on columns, similar to Ascon-p and Gaston.

The linear layer λ typically consists of mixing layers that mix the bits (or bytes) and shuffle layers that move bits. Here we are interested in a linear layer which has the following structure: $\lambda = \rho_{east} \circ \theta \circ \rho_{west}$. θ is a mixing layer that ensures a bit at its output depends on multiple bits at its input. ρ_{east} and ρ_{west} are two ShiftRow-like shuffles that move bits which are close to each other to positions that are far from each other.

For the round function R that is iterated multiple times, we would like to resist differential (DC) and linear cryptanalysis (LC). To resist differential and linear attacks, the round function needs to avoid high-probability differential trails and linear trails with high correlation contributions.

In this section, we recall the fundamental concepts of difference and linear propagation and list some performance metrics for θ and λ such as the branch number and the branch histogram.

2.1 Differential and Linear Cryptanalysis

Biham and Shamir's differential cryptanalysis [BS91] is one of the two main statistical cryptanalysis techniques. Consider a function f defined over $\mathbb{F}_2^n$. An input difference a to the function f and the corresponding output difference b form a differential over f. The differential probability (DP) is defined as the proportion of all possible input pairs with a difference of a that results in a difference of b after applying the function f to these pairs.

$$\mathrm{DP}(a,b) = \frac{\#\left\{x \in \mathbb{F}_2^n \mid f(x) \oplus f(x \oplus a) = b\right\}}{2^n}.$$

The *differential weight* $\mathrm{w_r}$ of a differential relates to its DP as $\mathrm{DP} = 2^{-\mathrm{w_r}}$. A differential over the round function R is referred to as a round differential. These round differentials can be linked together to form a *differential trail*. An r-round differential trail T is defined by a sequence of difference patterns before and after each round, denoted as $\left(t^0, t^1, \ldots, t^r\right)$. Calculating the DP of a trail can be quite challenging, so it is often approximated using its expected differential probability (EDP). The EDP of a differential trail is obtained by multiplying the DP values of its individual round differentials, assuming these round differentials operate independently:

$$\mathrm{EDP}(T) = \prod_{0<i\leq r} \mathrm{DP}\left(t^{i-1}, t^i\right).$$

The differential weight of a trail is the total of the differential weights of its round differentials if the round differentials act independently. Consequently, the relationship between the differential weight $\mathrm{w_r}(T)$ and the EDP is expressed as $2^{-\mathrm{w_r}(T)} = \mathrm{EDP}(T)$.

Matsui's linear cryptanalysis [BS91] is the other important statistical attack and shares many concepts with differential cryptanalysis. It considers masks α for inputs and β for outputs instead of differences. (α, β) is called a linear approximation over f. The correlation of this approximation is determined by the probability p over all inputs x that the linear functions defined by α and β are equal, namely $2p - 1$:

$$\mathrm{C}(\alpha, \beta) = \frac{\#\left\{x \in \mathbb{F}_2^n \mid \alpha^{\mathrm{T}} x + \beta^{\mathrm{T}} f(x) = 0\right\}}{2^{n-1}} - 1.$$

A linear approximation over the round function R is called a round linear approximation. These round linear approximations can be connected to form a *linear trail*. An r-round linear trail T is characterized by a sequence of masks before and after each round, represented as $\left(t^0, t^1, \ldots, t^r\right)$. The correlation contribution C of a trail is the product of the correlations of its round linear approximations:

$$\mathrm{C}(T) = \prod_{0<i\leq r} \mathrm{C}\left(t^{i-1}, t^i\right).$$

The *correlation weight* w_c of a linear trail relates to its squared correlation as $\mathrm{C}^2 = 2^{-\mathrm{w}_c}$.

Remark. For a block cipher, if its round keys are independent of each other, we can use $\mathrm{EDP}(T)$ and $\mathrm{C}(T)$ to calculate the probabilities of differential and linear trails, respectively [LMM91]. In this paper, we analyze the differential and linear trails of permutations. Although these permutations do not have alternating round keys, $\mathrm{EDP}(T)$ and $\mathrm{C}(T)$ are still commonly used to estimate the approximation probabilities of these trails. For example, for the permutation Ascon, the approximate probabilities of many existing differential or linear trails have been obtained using $\mathrm{EDP}(T)$ and $\mathrm{C}(T)$ [DEMS15, GPT21, EME22].

2.2 Diffusion Metrics Related to Differences

In this section, we discuss diffusion metrics for the propagation of differences. We refer to the nonzero bits in the differential state as active bits, and the nonzero columns as active S-boxes. To resist differential analysis, we hope for better diffusion of the active bits and active S-boxes of difference in the linear layer λ. The differential branch number introduced in [Dae95] and widely disseminated through [JV02], is an important metric for the diffusion power of mixing layers. As summarized in [HDRM23], we define the following concepts. The differential state is defined as the difference between the states of the two inputs.

Definition 1 (Bit branch number of a differential state [HDRM23])**.** The bit branch number of differential state $\boldsymbol{D}$ with respect to linear layer L is the sum of the bit weight of $\boldsymbol{D}$ and that of L:

$$\mathcal{B}_{b,L}(\boldsymbol{D}) = w_b(\boldsymbol{D}) + w_b(L(\boldsymbol{D})),$$

where $w_b(\boldsymbol{D})$ is the number of active bits in state $\boldsymbol{D}$.

Definition 2 (Column branch number of a differential state [HDRM23])**.** The column branch number of differential state $\boldsymbol{D}$ with respect to linear layer L is the sum of the column weight of $\boldsymbol{D}$ and that of L:

$$\mathcal{B}_{c,L}(\boldsymbol{D}) = w_c(\boldsymbol{D}) + w_c(L(\boldsymbol{D})),$$

where $w_c(\boldsymbol{D})$ is the number of active columns in state $\boldsymbol{D}$.

The differential branch numbers of a linear layer L are the minimum of the corresponding branch number over all nonzero differential states. The bit and column differential branch numbers of L are given by:

$$\mathcal{B}_b(L) = \min_{\boldsymbol{D} \in \mathcal{D} \setminus \{0\}} \mathcal{B}_{b,L}(\boldsymbol{D}) \quad \text{and} \quad \mathcal{B}_c(L) = \min_{\boldsymbol{D} \in \mathcal{D} \setminus \{0\}} \mathcal{B}_{c,L}(\boldsymbol{D}).$$

Although the differential branch number measures the diffusion power of a permutation, it provides only limited information. We want to determine the diffusion power when many linear layers have the same branch number. So, we recall a more detailed concept *branch histograms.*

Definition 3 (Bit branch histogram [HDRM23])**.** The bit branch histogram of L is the histogram indicating the number of states per bit branch number:

$$H_{b,L}(w) = \# \left\{ \boldsymbol{D} \in \mathcal{D} \text{ with } \mathcal{B}_{b,L}(\boldsymbol{D}) = w \right\}.$$

The bit branch histogram is the appropriate measure for a mixing layer like θ. We hope to have a few states with low bit branch numbers for θ to have good diffusion.

Definition 4 (Column branch histogram [HDRM23])**.** The column branch histogram of L is the histogram indicating the number of states per column branch number:

$$H_{c,L}(w) = \# \left\{ \boldsymbol{D} \in \mathcal{D} \text{ with } \mathcal{B}_{c,L}(\boldsymbol{D}) = w \right\}.$$

The column branch histogram is the appropriate measure for a linear layer λ. A linear layer with good diffusion properties should have few states with low column branch numbers, i.e., it has a branch histogram with a low left tail.

2.3 Diffusion Metrics Related to Masks

As demonstrated in [Dae95], the linear branch number of a linear mapping defined by a matrix L is equivalent to the differential branch number of L^T, which is the transpose of L. Hence, when discussing the linear branch numbers of a linear mapping, we refer to the (differential) branch number of its transpose. The mask state is defined as the value of the input mask.

Since we define the linear branch number of a mask state $\boldsymbol{D}$ with respect to a linear layer L as the sum of the weights of $\boldsymbol{D}$ and that of $\alpha = L^T(\boldsymbol{D})$, the bit and column linear branch numbers of a mask state $\boldsymbol{D}$ are given by:

$$\mathcal{B}_{b,L^{\mathrm{T}}}(\boldsymbol{D}) = w_b(\boldsymbol{D}) + w_b\left(L^{\mathrm{T}}(\boldsymbol{D})\right) \quad \text{and} \quad \mathcal{B}_{c,L^{\mathrm{T}}}(\boldsymbol{D}) = w_c(\boldsymbol{D}) + w_c\left(L^{\mathrm{T}}(\boldsymbol{D})\right).$$

Similarly, the linear bit and column branch numbers of a mapping L are defined by

$$\mathcal{B}_b\left(L^{\mathrm{T}}\right) = \min_{\boldsymbol{D}\in\mathcal{D}\setminus\{0\}} \mathcal{B}_{b,L^{\mathrm{T}}}(\boldsymbol{D}) \quad \text{and} \quad \mathcal{B}_c\left(L^{\mathrm{T}}\right) = \min_{\boldsymbol{D}\in\mathcal{D}\setminus\{0\}} \mathcal{B}_{c,L^{\mathrm{T}}}(\boldsymbol{D}).$$

The linear bit branch histogram and the linear column branch histogram of a mapping L are defined by

$$H_{b,L^T}(w) = \#\left\{\boldsymbol{D} \in \mathcal{D} \text{ with } \mathcal{B}_{b,L^T}(\boldsymbol{D}) = w\right\},$$
$$H_{c,L^T}(w) = \#\left\{\boldsymbol{D} \in \mathcal{D} \text{ with } \mathcal{B}_{c,L^T}(\boldsymbol{D}) = w\right\}.$$

2.4 Column Parity Mixers

The column parity mixer (we call it CPM for short) was first used in Keccak. At FSE 2018, Stoffelen and Daemen [SD18] proposed a formal definition of column parity mixers. The column parity mixer is a linear mapping that sends a state matrix $\boldsymbol{A} \in \mathbb{F}_2^{m\times n}$ to another $\boldsymbol{B} \in \mathbb{F}_2^{m\times n}$, which is parameterized by 2 rotation constants u and r. Let A_i and B_i be the i-th row of $\boldsymbol{A}$ and $\boldsymbol{B}$, then the operation of the CPM is represented by

$$\begin{aligned} B_i &\leftarrow A_i + (E \lll u) \text{ for } 0 \le i < m \\ &\qquad \text{with } E \leftarrow (P + (P \lll r)) \text{ and } P \leftarrow \sum_{k=0}^{m-1} A_k. \end{aligned} \tag{1}$$

Here P is the column parity which is the bitwise sum (XOR) of all rows, $\lll$ denotes the left cyclic shift operation, and E is calculated by adding P and a shifted copy of P. Then E is added to each row after shifting over the offset u. The CPM has a computational cost of 2 XORs per bit. Its bitwise differential branch number and bitwise linear branch number are both 4.

Definition 5 (The kernel of a CPM)**.** For a CPM, the space that includes states that have $E = 0$ in (1) is its kernel.

The states in the kernel are crucial as they pass the mixing layer "for free". For a CPM, the states whose columns have even parity are obviously in the kernel. States in the kernel that have no active bits except for two active bits on one column will determine the differential branch number of the CPM, i.e., 4.

2.5 Circulant Twin Column Parity Mixers and Gaston

The circulant twin column parity mixer, abbreviated as the TCPM for convenience, was designed by Hirch et al. at CRYPTO 2023 [HDRM23]. It is a generalization of the column parity mixer. The TCPM is a linear mapping that sends a state matrix $\boldsymbol{A} \in \mathbb{F}_2^{m\times n}$ to another $\boldsymbol{B} \in \mathbb{F}_2^{m\times n}$, which is parameterized by $3 + m$ rotation constants u, r, s and $t_0, t_1, \ldots, t_{m-1}$. Let A_i and B_i be the i-th row of $\boldsymbol{A}$ and $\boldsymbol{B}$, then the operation of the TCPM is represented by

$$\begin{aligned} B_i &\leftarrow A_i + (E \lll u) \text{ for } 0 \le i < m \\ &\qquad \text{with } E \leftarrow (P + (P \lll r)) + (Q + (Q \lll s)), \\ &\qquad \text{and } P \leftarrow \sum_{k=0}^{m-1} A_k \text{ and } Q \leftarrow \sum_{k=0}^{m-1} (A_k \lll t_k). \end{aligned} \tag{2}$$

The twin CPM has a computational cost of $(3 + \frac{1}{m})$ XORs per bit. Its bitwise differential branch number is 12 and its bitwise linear branch number is 4.

Definition 6 (The kernel of a TCPM)**.** For a TCPM, the space that includes states that have $E = 0$ in (2) is its kernel.

For a TCPM, the nonzero state whose differential branch number is the smallest is in the kernel [HDRM23, Lemma 5]. Such a state has two active bits in three columns and none in other columns (to make $P = 0$). Moreover, the shifted state of this state has two active bits in three columns and none in other columns (to make $Q = 0$).

Compared with the CPM, the TCPM has a better differential branch number mainly because the minimum bits of the nonzero state in the kernel are more than the CPM.

Hirch et al. gave a concrete iterated permutation using the TCPM called Gaston [HDRM23]. It operates on a state of 5×64 bits and uses χ operating $(A_j + \overline{A_{j+1}} \cdot A_{j+2})$ on columns for its non-linear layer. Gaston requires as few bitwise operations as the NIST lightweight standard Ascon, but it has much higher weights for the best 3-round differential and linear trails. Specification of Gaston is provided in Appendix D.1.

3 Linear Analysis of the TCPM

Although the TCPM has a large differential branch number, its linear branch number is only 4 which is the same as that of the CPM. In this section, we first show why the linear branch number of the TCPM is only 4. We then report new linear trails of Gaston and compare them with Ascon.

3.1 Analyzing the Linear Branch Number of the TCPM

Since the linear branch number of a mixing layer is equal to the branch number of its transpose, we focus on the transpose of the TCPM. Recall that a TCPM is applied to an $m \times n$ state, in this section, we regard this state as an mn-dimensional column vector $\boldsymbol{A} = [A_0, A_1, \ldots, A_{m-1}]^T$, where $A_i = [a_{ni}, a_{ni+1}, \ldots, a_{ni+n-1}]^T$ for $0 \le i < m$ is the i-th row of the state. Let $\boldsymbol{I}_n, \boldsymbol{\rho}_x \in \mathbb{F}_2^{n\times n}$ represent the identity matrix of dimension n and the matrix of a cyclic rotation of x bits to the left, respectively. We have

$$\boldsymbol{\rho}_x A_i = A_i \lll x = [a_{ni+x}, a_{ni+x+1}, \ldots, a_{ni+n-1}, a_{ni}, \ldots, a_{ni+x-1}]^T, x \in \mathbb{Z}_n.$$

Now, the output $\boldsymbol{B}$ of the TCPM can be expressed as

$$B_i \leftarrow A_i + \sum_{k=0}^{m-1} A_k(\boldsymbol{\rho}_u + \boldsymbol{\rho}_{u+r} + \boldsymbol{\rho}_{u+t_k} + \boldsymbol{\rho}_{u+t_k+s}),$$

where $u, r, s, t_0, \ldots, t_{m-1}$ are the rotation constants of this TCPM. Assume $\boldsymbol{l}_0, \ldots, \boldsymbol{l}_{m-1} \in \mathbb{F}_2^{n\times n}$ are $n \times n$ matrices satisfying

$$\boldsymbol{l}_k = \boldsymbol{\rho}_u + \boldsymbol{\rho}_{u+r} + \boldsymbol{\rho}_{u+t_k} + \boldsymbol{\rho}_{u+t_k+s}, \text{ where } 0 \le k < m.$$

Then the TCPM parameterized by $u, r, s, t_0, \ldots, t_{m-1}$ can be represented as an $mn \times mn$ matrix $\boldsymbol{L}_0$ given by

$$\boldsymbol{L}_0 = \begin{bmatrix} \boldsymbol{l}_0 + \boldsymbol{I}_n & \boldsymbol{l}_1 & \boldsymbol{l}_2 & \ldots & \boldsymbol{l}_{m-1} \\ \boldsymbol{l}_0 & \boldsymbol{l}_1 + \boldsymbol{I}_n & \boldsymbol{l}_2 & \ldots & \boldsymbol{l}_{m-1} \\ \boldsymbol{l}_0 & \boldsymbol{l}_1 & \boldsymbol{l}_2 + I_n & \ldots & \boldsymbol{l}_{m-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{l}_0 & \boldsymbol{l}_1 & \boldsymbol{l}_2 & \ldots & \boldsymbol{l}_{m-1} + \boldsymbol{I}_n \end{bmatrix}, \quad (3)$$

and the transpose of $\boldsymbol{L}_0$ can be expressed as

$$\boldsymbol{L}_0^T = \begin{bmatrix} \boldsymbol{l_0} + \boldsymbol{I}_n & \boldsymbol{l_1} & \dots & \boldsymbol{l}_{m-1} \\ \boldsymbol{l}_0 & \boldsymbol{l}_1 + \boldsymbol{I}_n & \dots & \boldsymbol{l}_{m-1} \\ \boldsymbol{l}_0 & \boldsymbol{l}_1 & \dots & \boldsymbol{l}_{m-1} \\ \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{l}_0 & \boldsymbol{l}_1 & \dots & \boldsymbol{l}_{m-1} + \boldsymbol{I}_n \end{bmatrix}^T = \begin{bmatrix} \boldsymbol{l_0}^T + \boldsymbol{I}_n & \boldsymbol{l_0}^T & \dots & \boldsymbol{l}_0^T \\ \boldsymbol{l}_1^T & \boldsymbol{l}_1^T + \boldsymbol{I}_n & \dots & \boldsymbol{l}_1^T \\ \boldsymbol{l}_2^T & \boldsymbol{l}_2^T & \dots & \boldsymbol{l}_2^T \\ \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{l}_{m-1}^T & \boldsymbol{l}_{m-1}^T & \dots & \boldsymbol{l}_{m-1}^T + \boldsymbol{I}_n \end{bmatrix}, \tag{4}$$

where $\boldsymbol{l}_0^T, \dots, \boldsymbol{l}_{m-1}^T$ are $n \times n$ matrices satisfying:

$$\boldsymbol{l}_k^T = \boldsymbol{\rho}_u^T + \boldsymbol{\rho}_{u+r}^T + \boldsymbol{\rho}_{u+t_k}^T + \boldsymbol{\rho}_{u+t_k+s}^T, \text{ where } 0 \leq k < m.$$

Hence, the transpose of the TCPM can be re-written as

$$B_i \leftarrow A_i + \sum_{k=0}^{m-1} A_k(\boldsymbol{\rho}_u^T + \boldsymbol{\rho}_{u+r}^T + \boldsymbol{\rho}_{u+t_i}{}^T + \boldsymbol{\rho}_{u+t_i+s}{}^T),$$

Equivalently, the transpose of TCPM can also be expressed as

$$\begin{aligned} B_i \leftarrow A_i + (E_i \ggg u) \text{ for } 0 \leq i < m \\ \text{with } E_i \leftarrow (P + (P \ggg r)) + (Q_i + (Q_i \ggg s)), \\ \text{and } P \leftarrow \sum_{k=0}^{m-1} A_k \text{ and } Q_i \leftarrow \sum_{k=0}^{m-1} (A_k \ggg t_i)\,. \end{aligned}$$

As a result, the rotation constant t_i (colored in red) of each row in Q_i are independent from the index of rows, it becomes a shift of P, i.e.,

$$Q_i \leftarrow \sum_{k=0}^{m-1}(A_k \ggg t_i) = (\sum_{k=0}^{m-1} A_k) \ggg t_i = P \ggg t_i. \tag{5}$$

To have the input and output states of the transpose of the TCPM have the least active bits, we only need to have $P = 0$ by setting 2 active bits for one certain column of the input states. Therefore, the linear branch number of the TCPM is 4.

From the above analysis, to prevent the linear branch number from being 4, we only need to have t_i related to the index of the rows. In the following, we will show that Gaston which uses the TCPM as its mixing layer has a weaker resistance against the linear cryptanalysis than Ascon for 4, 5, and 6 rounds, so if we want a stronger permutation against differential and linear attacks, increasing the linear branch number (while keeping the differential branch number) of the TCPM is an interesting topic. In Section 4, we will introduce a new strategy to design a mixing layer that has both 12 differential and linear branch number.

3.2 Linear Trails of Gaston

In this section, we present new linear trails of Gaston and show that Gaston's linear behavior seems worse than that of Ascon when the number of rounds increases.

Iterative trails. By setting $P = 0$ at the first round, we find an iterative linear trail (see Appendix D.2) for 2 rounds with squared correlation 2^{-48}. Extending this 2-round trail, we can construct l rounds (with $l \geq 2$) iterative linear trail with squared correlation $2^{-2 \cdot 12 \cdot l}$. This means for $l = 4, 5, 6$ rounds, there exist linear trails with squared correlations 2^{-96}, 2^{-120} and 2^{-144}, respectively.

A 4-round trail with squared correlation 2^{-82}. Having seen that $P = 0$ at the first round gives an iterative trail with low weight, we investigate whether setting $P = 0$ in further rounds gives even better trails. We find that by setting $P = 0$ for the first three rounds, we obtain a 4-round linear trail with squared correlation 2^{-82} (see Appendix D.3). This improves the previous squared correlation, i.e., 2^{-96} of 4-round iterative trail by a factor of 2^{14}.

Comparison with linear trails of Ascon. In Table 2, we provide a comparison of the best known linear trails of Gaston and Ascon. Notice that the best-known squared correlation for 5-round linear trail of Ascon is 2^{-184}. Hence, we deduce that the best squared correlation for 6-round linear trail should generally be smaller than 2^{-184}. In the case of Gaston, the best squared correlation for 6-round linear trail is 2^{-144} which is much higher than the squared correlation of 5-round linear trail of Ascon. Accordingly, we claim Gaston has a weaker resistance against linear cryptanalysis than Ascon for 6 rounds as well. In summary, Gaston's linear trails are worse than Ascon for more than 3 rounds, a crucial observation which was missed by the designers.

Table 2: A comparison of linear trails of Gaston and Ascon

Rounds	Best known squared correlation of	
	Gaston	Ascon
3	2^{-34} [HDRM23]	2^{-28} [DEM15]
4	2^{-82} (Appendix D.3)	2^{-98} [DEM15]
5	2^{-120} (extend Appendix D.2 by 3 rounds)	2^{-184} [MR22]
6	2^{-144} (extend Appendix D.2 by 4 rounds)	-

4 Mixing Layers with Differential and Linear Branch Numbers 12

In this section, we explain the design of a mixing layer which has both differential and linear branch numbers equal to 12. We then introduce a specialized variant of this mixing layer called the *symmetric twin column parity mixer*. Next, we present some of its properties, along with their proof processes, and provide a detailed analysis of its diffusion properties in Chapter 5.

4.1 Modified Circulant Twin Column Parity Mixer

In Section 3.1, we have proved that the linear branch number of the TCPM is 4 because of Eqn. (5). To avoid this situation, elements from the same row in matrix 4 ($\boldsymbol{L}_0^T$) should not have the same $\boldsymbol{l}_k$. This is equivalent to the fact that elements from the same column of the matrix 3 ($\boldsymbol{L}_0$) should not have the same $\boldsymbol{l}_k$. So, we construct a matrix $\boldsymbol{L}_1$

$$\boldsymbol{L}_1 = \begin{bmatrix} \boldsymbol{l}_{0,0}+\boldsymbol{I}_n & \boldsymbol{l}_{0,1} & \boldsymbol{l}_{0,2} & \cdots & \boldsymbol{l}_{0,m-1} \\ \boldsymbol{l}_{1,0} & \boldsymbol{l}_{1,1}+\boldsymbol{I}_n & \boldsymbol{l}_{1,2} & \cdots & \boldsymbol{l}_{1,m-1} \\ \boldsymbol{l}_{2,0} & \boldsymbol{l}_{2,1} & \boldsymbol{l}_{2,2}+\boldsymbol{I}_n & \cdots & \boldsymbol{l}_{2,m-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{l}_{m-1,0} & \boldsymbol{l}_{m-1,1} & \boldsymbol{l}_{m-1,2} & \cdots & \boldsymbol{l}_{m-1,m-1}+\boldsymbol{I}_n \end{bmatrix}, \quad (6)$$

where each of its elements have different $\boldsymbol{l}_{i,j}$ ($\boldsymbol{l}_{i,j}$ is a $n \times n$ matrix) such that

$$\boldsymbol{l}_{i,j} = \boldsymbol{\rho}_u + \boldsymbol{\rho}_{u+r} + \boldsymbol{\rho}_{u+t_i+b_j} + \boldsymbol{\rho}_{u+t_i+b_j+s}, \text{ where } 0 \leq i,j < m. \quad (7)$$

In Eqn. (7), t_i ensures that each row in Q operation is shifted differently for $\boldsymbol{L}_1$ and b_j does the same thing for $\boldsymbol{L}_1^T$. Notice that $\boldsymbol{L}_1$ and $\boldsymbol{L}_1^T$ are both modified TCPMs. By selecting the appropriate parameters, the differential and linear branch number of $\boldsymbol{L}_1$ can be both 12. The matrix $\boldsymbol{L}_1$ can be expressed as

$$\begin{aligned} &B_i \leftarrow A_i + (E_i \lll u) \text{ for } 0 \leq i < m \\ &\qquad\qquad\qquad \text{with } E_i \leftarrow (P + (P \lll r)) + (Q_i + (Q_i \lll s)), \\ &\qquad\qquad\qquad \text{and } P \leftarrow \sum_{k=0}^{m-1} A_i \text{ and } Q_i \leftarrow \sum_{k=0}^{m-1} (A_k \lll (t_k + b_i)) \,. \end{aligned}$$

It can also be expressed in the following form

$$\begin{aligned} B_i \leftarrow & A_i + ((E + (F \lll b_i)) \lll u) \text{ for } 0 \leq i < m \\ & \text{with } E \leftarrow (P + (P \lll r)),\ F \leftarrow (Q + (Q \lll s)), \\ & \text{and } P \leftarrow \sum_{k=0}^{m-1} A_k \text{ and } Q \leftarrow \sum_{k=0}^{m-1} (A_k \lll t_k) \,. \end{aligned} \tag{8}$$

Similarly, $\boldsymbol{L}_1^T$ can be written as

$$\begin{aligned} B_i \leftarrow & A_i + ((E + (F \ggg t_i)) \ggg u) \text{ for } 0 \leq i < m \\ & \text{with } E \leftarrow (P + (P \ggg r)),\ F \leftarrow (Q + (Q \ggg s)), \\ & \text{and } P \leftarrow \sum_{k=0}^{m-1} A_k \text{ and } Q \leftarrow \sum_{k=0}^{m-1} (A_k \ggg b_k) \,. \end{aligned} \tag{9}$$

Comparing Eqn. (8) with Eqn. (2), we observe that the modified TCPM is essentially the original TCPM with an additional set of cyclic shift parameters $b_0, \ldots, b_{m-1}$. Note that by setting $b_0, \ldots, b_{m-1}$ as zero, we retrieve the TCPM.

Consequently, at the cost of adding some shift offsets, the linear branch number for the modified TCPM can achieve 12 (increased from the original 4 of the TCPM) by selecting appropriate parameters. However, the computational cost increases from the original $(3 + \frac{1}{m})$ XORs per bit to 4 XORs per bit because some values cannot be reused during the linear layer computation. As a result, both the hardware implementation area and power consumption increase compared to the original TCPM. Fortunately, since only circulant shift operations are added, the maximum XOR depth of the modified TCPM remains unchanged at $\lceil \log_2 (4m + 1) \rceil$, ensuring that the hardware latency is similar to that of the original TCPM. In terms of software implementation, the modified TCPM implementation is slightly slower than the original TCPM due to the additional cyclic shifts. These points will be confirmed in Section 7.3, where we compare the efficiency of Gaston and Gaston-S (which replaces the original TCPM in Gaston with the modified TCPM) implementations.

4.2 The Symmetric Circulant Twin Column Parity Mixer

Although the differential branch number and linear branch number for the modified TCPM are both 12, we would like to have its differential branch histogram and linear branch histogram to be both well-distributed. For an $m \times n$ two-dimensional state, the modified TCPM has $(2m + 3)$ shift offset parameters: the two folding offsets r and s, the addition offset u, and $2m$ offsets $t_0, \ldots, t_{m-1}, b_0, \ldots, b_{m-1}$ for computing parity Q.

For large-dimensional states, the parameter space is too huge. Moreover, a set of parameters needs to be chosen considering differential and linear security simultaneously. These considerations make the parameters selection process challenging. To solve this problem, we introduce the symmetric TCPM. It is a special kind of the modified TCPM

(Section 4.1) where we set $b_k = t_k$ for $0 \le k < m$. We call it the symmetric TCPM for short.

The symmetric TCPM applied to an $m \times n$ state can be expressed as

$$\begin{aligned} A_i \leftarrow & A_i + ((E + (F \lll t_i)) \lll u) \text{ for } 0 \le i < m \\ & \text{with } E \leftarrow (P + (P \lll r)),\ F \leftarrow (Q + (Q \lll s)), \\ & \text{and } P \leftarrow \sum_{k=0}^{m-1} A_k \text{ and } Q \leftarrow \sum_{k=0}^{m-1} (A_k \lll t_k)\,. \end{aligned} \tag{10}$$

Similar to the modified TCPM in Section 4.1, its differential branch number and linear branch number are both 12 and its computational cost is 4 XORs per bit. Moreover, the symmetric TCPM has some special properties which we discuss in Theorems 1 to 4.

Theorem 1. *Let $\boldsymbol{L}$ be a symmetric TCPM applied to an $m \times n$ state and for all $0 \le i, j < m$, $\{0, u, u+r, u+t_i+t_j, u+t_i+t_j+s\}$ are not pairwise equal. Then $\boldsymbol{L}$ has $(4m+1)$ ones in each row or column.*

Proof. It is easy to deduce from Eqn. (6). □

Theorem 2. *Let $\boldsymbol{L}$ be a symmetric TCPM applied to an $m \times n$ state. Then $\boldsymbol{L}^T$ is also a symmetric TCPM.*

Proof. See Appendix A.1. □

Theorem 3. *Let $\boldsymbol{L}$ be an $m \times n$ dimensional symmetric TCPM. Suppose there exists a differential trail $\boldsymbol{D} \xrightarrow{\boldsymbol{L}} \boldsymbol{O}$, then there exists a corresponding linear trail $\boldsymbol{\alpha} \xrightarrow{\boldsymbol{L}} \boldsymbol{\beta}$, where $\beta_{i,n-j} = D_{i,j}$ and $\alpha_{i,n-j} = O_{i,j}$, for $0 \le i < m, 0 \le j < n$.*

Proof. Since $\boldsymbol{D} \xrightarrow{\boldsymbol{L}} \boldsymbol{O}$, according to Eqn. (10), the propagation of difference can be expressed as

$$\begin{aligned} O_{i,j} \leftarrow & D_{i,j} + \sum_{k=0}^{m-1} D_{k,(j+u) \bmod n} + \sum_{k=0}^{m-1} D_{k,(j+u+r) \bmod n} \\ & + \sum_{k=0}^{m-1} D_{k,(j+t_k+t_i) \bmod n} + \sum_{k=0}^{m-1} D_{k,(j+t_k+t_i+s) \bmod n}. \end{aligned} \tag{11}$$

By the definition of linear masks, we have

$$\boldsymbol{\beta} \cdot \boldsymbol{y} = \boldsymbol{\beta} \cdot (\boldsymbol{L}\boldsymbol{x}) = \left(\boldsymbol{L}^T \boldsymbol{\beta}\right) \cdot \boldsymbol{x} \rightarrow \boldsymbol{\alpha} = \boldsymbol{L}^T \boldsymbol{\beta}$$

Then, following Eqn. (9), the propagation of linear masks $\boldsymbol{\alpha} \xrightarrow{L} \boldsymbol{\beta}$ can be expressed as

$$\begin{aligned} \alpha_{i,j} \leftarrow & \beta_{i,j} + \sum_{k=0}^{m-1} \beta_{i,(j-u) \bmod n} + \sum_{k=0}^{m-1} \beta_{i,(j-u-r) \bmod n} \\ & + \sum_{k=0}^{m-1} \beta_{i,(j-u-t_k-t_i) \bmod n} + \sum_{k=0}^{m-1} \beta_{i,(j-u-t_k-t_i-s) \bmod n}. \end{aligned} \tag{12}$$

Plugging the expressions $\beta_{i,n-j} = D_{i,j}$ and $\alpha_{i,n-j} = O_{i,j}$ into Eqn. (11) we have

$$\begin{aligned} \alpha_{i,n-j} \leftarrow & \beta_{i,n-j} + \sum_{k=0}^{m-1} \beta_{k,(n-(j+u)) \bmod n} + \sum_{k=0}^{m-1} \beta_{k,(n-(j+u+r)) \bmod n} \\ & + \sum_{k=0}^{m-1} \beta_{k,(n-(j+t_k+t_i)) \bmod n} + \sum_{k=0}^{m-1} \beta_{k,(n-(j+t_k+t_i+s)) \bmod n}. \end{aligned}$$

Let $j' = n - j$, then we have

$$\begin{aligned}\alpha_{i,j'} \leftarrow & \beta_{i,j'} + \sum_{k=0}^{m-1} \beta_{i,(j'-u) \bmod n} + \sum_{k=0}^{m-1} \beta_{i,(j'-u-r) \bmod n} \\ & + \sum_{k=0}^{m-1} \beta_{i,(j'-u-t_k-t_i) \bmod n} + \sum_{k=0}^{m-1} \beta_{i,(j'-u-t_k-t_i-s) \bmod n}.\end{aligned}$$

Obviously, the above satisfies Eqn. (12) as well which completes the proof. □

Theorem 4. *The bit differential branch histogram of a symmetric TCPM is the same as its bit linear branch histogram.*

Proof. It is easily derived from Theorem 3, and the detailed process is provided in Appendix A.2. □

4.3 Cyclic Row Shifts

The symmetric TCPM is a mixing layer that ensures a bit at its output depends on multiple bits at its input. For the complete linear layer, we need two ShiftRow-like shuffle layers ρ_{east} and ρ_{west} to move bits which are close to each other to positions that are far from each other.

Let ρ_{east} and ρ_{west} are two bit shuffles that cyclically shift the bits within rows:

$$\rho_{east} : A_i \leftarrow (A_i \lll w_i), \text{ for } 0 \le i < m,$$

$$\rho_{west} : A_i \leftarrow (A_i \lll e_i), \text{ for } 0 \le i < m.$$

Assume the linear layer $\lambda = \rho_{east} \circ \boldsymbol{L} \circ \rho_{west}$. Then by selecting appropriate parameters, the column differential branch number and the column linear branch number of λ could be both 12.

Based on Theorem 3 and Theorem 4, a good choice is to set $\rho_{east} = \rho_{west}$. In practice, the optimal parameter combinations, determined by balancing linear and differential security, consistently satisfy $\rho_{east} = \rho_{west}$. The linear layer λ has the following properties when $\rho_{east} = \rho_{west}$.

Theorem 5. *Let $\lambda = \rho \circ \boldsymbol{L} \circ \rho$ where $\boldsymbol{L}$ is a symmetric TCPM and $\rho = (w_0, \ldots, w_{m-1})$ is a bit shuffle. If there is a differential trail $\boldsymbol{D} \xrightarrow{\lambda} \boldsymbol{O}$, then there must be a corresponding linear trail $\boldsymbol{\alpha} \xrightarrow{\lambda} \boldsymbol{\beta}$, where $\beta_{i,n-j} = D_{i,j}$ and $\alpha_{i,n-j} = O_{i,j}$, for $0 \le i < m, 0 \le j < n$.*

Proof. See Appendix A.3. □

Theorem 6. *Assume $\lambda = \rho \circ \boldsymbol{L} \circ \rho$ where $\boldsymbol{L}$ is an $m \times n$ dimensional symmetric TCPM and $\rho = (w_0, \ldots, w_{m-1})$ is a bit shuffle. The column differential branch histogram of λ is the same as its column linear branch histogram.*

Proof. The proof is similar to Theorem 4 and we omit it. □

5 Diffusion Properties of the Symmetric TCPM

In this section, we analyze the diffusion behavior of the symmetric TCPM. Our goal is to ensure that the branch number of the symmetric TCPM is at least 12 and that there are low upper left tails in its branch histogram. Thanks to Theorem 4, we only need to analyze either the differential or linear diffusion properties of the symmetric TCPM. We choose the differential properties for analysis. For the convenience of analysis, we express the difference state as a polynomial.

5.1 Polynomial Representation

Let the symmetric TCPM be applied to an $m \times n$ state. We denote the input difference state in terms of polynomials by $\boldsymbol{D} = (D_0(X), \ldots, D_{m-1}(X))$, where

$$D_i(X) = \sum_{j=0}^{n-1} d_{i,j} X^j \text{ for } 0 \le i < m \text{ and } 0 \le j < n$$

is the i-th row of the input difference state. Here $D_i(X) \in \mathbb{F}_2[X]$ and $\mathbb{F}_2[X]$ is a ring of binary polynomials. We consider the propagation of $D_i(X)$ through the symmetric TCPM which mainly includes two operations, the addition and the cyclic shift of rows. The addition of rows is just the addition of polynomials in $\mathbb{F}_2$. For cyclic shift of rows, we use $(X^r D_i(X)$ modulo $(1 + X^n))$ to represent $D_i(X) \lll r$. So, the row operations are operating in $\mathbb{F}_2$ modulo $(1 + X^n)$ and the input difference states are working in $(\mathbb{F}_2$ modulo $(1 + X^n))^m$.

In polynomial representation, the propagation of differences through the symmetric TCPM becomes:

$$\begin{aligned} O_i \leftarrow & D_i + X^u \left(\left(E + X^{t_i} F \right) \right) \text{ for } 0 \le i < m\,, \\ & \text{with } E \leftarrow (P + X^r P),\ F \leftarrow (Q + X^s Q), \\ & \text{and } P \leftarrow \sum_{k=0}^{m-1} D_k \text{ and } Q \leftarrow \sum_{k=0}^{m-1} X^{t_k} D_k. \end{aligned} \tag{13}$$

More concretely, we have

$$O_i \leftarrow D_i + X^u \sum_{k=0}^{m-1} (1 + X^r + X^{t_k + t_i} + X^{t_k + t_i + s}) D_k\,,\ \text{for } 0 \le i < m.$$

5.2 In the Kernel

When a state $\boldsymbol{D}$ satisfies $E + X^{t_i} F = 0$, $0 \le i < m$ in (13), the input difference equals the output difference. The branch number of this state is the number of active bits in $\boldsymbol{D}$ times two. We say these states are in the kernel.

Definition 7 (The kernel of a symmetric TCPM)**.** For a symmetric TCPM, the space that includes states that have $E + X^{t_i} F = 0$, $0 \le i < m$ in (13) is its kernel.

Notice that the states having $P = 0$ and $Q = 0$ in (13) are in the kernel. In fact, except for a few special cases, most states in the kernel have $P = 0$ and $Q = 0$. So, we focus on the states with $P = 0$ or $Q = 0$.

To ensure that the branch number of the symmetric TCPM is at least 12, we need to avoid states in the kernel with less than 6 active bits. Also, we want the number of states in the kernel with few active bits as small as possible to obtain low upper left tails in the branch histogram.

In this section, we adopt similar strategies as in [HDRM23, Section 4] to analyze the properties of states in kernel for the symmetric TCPM.

First, for the symmetric TCPM, when $n = 2^l$, by selecting the appropriate parameters, the states in the kernel will contain only an even number of active bits (Theorem 7).

Theorem 7. *Consider a symmetric TCPM applied to a two-dimensional $m \times n$ state. If $n = 2^l$ and $r + s$ mod 2 = 1, then the dimension of the kernel is $(m-1)n + 1$ and the states in the kernel only contain an even number of active bits.*

The proof is the same as [HDRM23, Corollary 1] and hence omitted.

Also, similar to the TCPM, no matter what shift offsets are selected, the kernel always contains states with 6 active bits. These states are called vortices.

Definition 8 (Vortice [HDRM23])**.** A state $\boldsymbol{D}$ is a vortex if it only has three active rows with indices in $\{i, j, k\} \subset \{0, 1, \ldots, m-1\}$ and

$$D_i = X^{t_j} + X^{t_k}, D_j = X^{t_k} + X^{t_i}, D_k = X^{t_i} + X^{t_j},$$

or if it is a shifted version of such a state.

Vortices have $P = D_i + D_j + D_k = 0$ and $Q = X^{t_i} D_i + X^{t_j} D_j + X^{t_k} D_k = 0$. Hence, it is in the kernel. There is no way to avoid vortices so the branch number of the symmetric TCPM cannot be more than 12.

Moreover, for some shift offsets, the kernel contains states with less than 6 active bits and we want to avoid that. These shift offsets are avoided for the TCPM by [HDRM23, Lemmas 6-8]. We use similar conditions to the symmetric TCPM to avoid low weight states in the kernel. We provide those lemmas and proofs in Appendix B.

5.3 Outside the Kernel

We now consider states outside the kernel with low branch numbers.

Theorem 8. *Consider a symmetric TCPM applied to a two-dimensional $m \times n$ state. If the number of active bits in $E + X^{t_i} F$ for a state is N_i, where $0 \leq i < m$, then the branch number of this state is at least $\sum_{i=0}^{m-1} N_i$.*

Proof. Let x and y be the number of active bits of the input state $\boldsymbol{D}$ and the output state $\boldsymbol{O}$, respectively. Since $O_i = E + X^{t_i} F + D_i$, we assume that the number of active bits colliding between $D_0, \ldots, D_{m-1}$ and $E + X^{t_0} F, \ldots, E + X^{t_{m-1}} F$ is z. Then we have $y \geq \sum_{i=0}^{m-1} N_i - z$. Consequently, the branch number $\mathcal{B} = y + x \geq \sum_{i=0}^{m-1} N_i - z + z = \sum_{i=0}^{m-1} N_i$ □

We discuss the active bits of $E + X^{t_i} F$ in terms of the active bits of E and F. Due to the folding operation, where $E = P + X^r P$ and $F = Q + X^s Q$, the number of active bits in E or F is always even.

If either E or F is zero while the other one has $2x$ active bits, the branch number of the state is at least $2xm$. The value of this lower bound increases as the number of rows (m) increases.

If both E and F are nonzero, let's assume E has $2x$ active bits and F has $2y$ active bits. In this case, the branch number of the state is at least $2xm + 2ym - 2z$, where z represents the number of colliding active bits in $E + X^{t_0} F, \ldots, E + X^{t_{m-1}} F$. When $x = 1$ and $y = 1$, the branch number $\mathcal{B}$ equals $2m + 2m - 2z$. While it is inevitable for some active bits of E and F to collide, we can try to avoid extremely unfavorable collision situations by selecting appropriate parameters.

Theorem 9. *If $\forall\ a \in [0, m)$, the elements in $\{t_a + t_0, t_a + t_1, \ldots, t_a + t_{m-1}\}$ are pairwise unequal and $r, s \neq 0$, then the branch number of states with only one active bit is at least $4m - 3$.*

Proof. See Appendix A.4. □

In the parameter selection process for given m and n, we record the states of the candidate parameters that make $2x + 2y$ smaller by experiments, and select the parameters that ensures that states have fewer colliding active bits in the mixing layer.

6 A Symmetric TCPM for 320-bit States

In this section, we present a concrete instance of the symmetric TCPM for state size $320 = 5 \times 64$ bits. We denote the linear layer as λ and the symmetric TCPM as θ. Accordingly, $\lambda = \rho_{east} \circ \theta \circ \rho_{west}$, where ρ_{east} and ρ_{west} are two ShiftRow-like shuffles.

For $m = 5$ and $n = 64$, there are 18 shift offsets for λ. These are $e_0, \dots, e_4$ and $w_0, \dots, w_4$ for ρ_{east} and ρ_{west}, respectively, and $t_0, \dots, t_4$ and u, r, s for θ. Each parameter has 64 possible values. We denote $\mathcal{R}_e = (e_0, \dots, e_4)$, $\mathcal{R}_w = (w_0, \dots, w_4)$ and $\mathcal{R}_\theta = (u, r, s, t_0, \dots, t_4)$. First, we select the set of $\mathcal{R}_\theta$ that exhibit low upper left tails in the bit branch histogram. Next, based on the selected $\mathcal{R}_\theta$, we choose $\mathcal{R}_e$ and $\mathcal{R}_w$ to achieve low upper left tails in the column branch histogram. According to Theorems 4 and 6, ensuring low upper left tails in the differential bit (column) branch histogram is sufficient for the linear bit (column) branch histogram to exhibit the same structure.

In the following, we explain our detailed rationale to find the proper choice of $\mathcal{R}_\theta$, $\mathcal{R}_w$ and $\mathcal{R}_e$.

6.1 Selecting the Offsets of $\mathcal{R}_\theta$

To select $\mathcal{R}_\theta$ offsets that result in low upper-left tails in the bit branch histogram, we aim for offsets that minimize the number of states with low branch numbers. Initially, we filter offsets that produce fewer states with low branch numbers in the kernel. Subsequently, we filter these offsets by selecting the ones that also have fewer states with low branch numbers outside the kernel. Since the value of offset u does not affect the distribution of states in the kernel, we begin by filtering the offsets $(r, s, t_0, \dots, t_4)$ through the following steps.

1. **Equivalence classes:** According to Theorem 7, we are required to have $(r + s) \bmod 2 = 1$. To reduce the search space, we fix $r = 1$ and set s to be an even number, which does not affect the differential and linear trail properties of a single linear layer.[1] Additionally, due to row order equivalence [HDRM23, Section 5], we arrange the values of row offsets t_j in an increasing order ($t_0 > t_1 > t_2 > t_3 > t_4$). This phase reduces the initial 2^{42} candidates of $(r, s, t_0, \dots, t_4)$ to approximately 2^{28} candidates.

2. **Filter using theoretical conditions:** Based on the parameters values conditions outlined in Theorem 1, Appendix B and Theorem 9, the 2^{28} candidates for $(r, s, t_0, \dots, t_4)$ could be reduced to 1295093, which is approximately $\approx 2^{25}$ candidates.

3. **Filter using experiments:** We record the candidates that have no in-kernel states with less than 6 active bits and the fewest in-kernel states with exactly 6 active bits. The minimum number of 6-bit states is 11 and there are $3743 \approx 2^{12}$ candidates. Next, we calculate the number of 8-bit states in the kernel (the next smallest number) for these candidates. Among them, 21 candidates have 140 8-bit states in the kernel. From these 21 candidates, we select those with the fewest number of 4-bit states with $P = 1, Q = 0$ or $P = 0, Q = 1$. The minimum number of such 4-bit states is 7, leading to 2 remaining candidates.

After the aforementioned filtering process, we identified two candidates. Then we select the appropriate u value for these two candidates to minimize the number of states with few active bits outside the kernel. The following outlines the filtration process.

[1] Fixing $r = 1$ may affect the differential and linear trail properties in more than 2 consecutive rounds. Here we only consider the differential and linear trail properties of a single linear layer due to the limitation of the search space.

1. **Filter using Theorem 1:** According to the theorem, the offsets should satisfy $u+r \neq 0; u \neq 0; u+t_i+t_j \neq 0; u+t_i+t_j+s \neq 0$, where $0 \leq i,j < 5$. Approximately half of the u candidates are eliminated based on these conditions.

2. **Filter using experiments:** We exhaustively check the remaining u candidates and identify those that distribute all active bits to different columns for states outside the kernel with 4 active bits. Subsequently, we choose the candidate that maximizes the distribution of active bits to different columns for states outside the kernel with 6 active bits, thereby determining the value of u.

After the above process, we obtain a set of parameters:

$$\mathcal{R}_\theta = (26, 36, 6, 10, 27, 41, 50).$$

Subsequently, we construct the 2-dimensional bit weight histogram associated with R_θ, as shown in Appendix A.5.

We report in Appendix A.6 the histogram of the difference between histograms $H_{b,L_1}(w)$ and $H_{b,L_2}(w)$, where L_1 is the symmetric TCPM we have chosen and L_2 is the TCPM used in Gaston.

6.2 Selecting the Offsets of $\mathcal{R}_e$ and $\mathcal{R}_w$

In the previous section, we selected the $\mathcal{R}_\theta$ to achieve a low upper left tail in the bit branch histogram. Given that the non-linear layer operates on columns, our objective is to minimize the number of states with a small column branch number. To achieve this, we carefully choose $\mathcal{R}_e$ and $\mathcal{R}_w$ parameters, ensuring that bits occupying the same column in the input or output state for θ-operation are moved across different columns. We proceed as follows.

1. **Filter using theoretical conditions:** Following Theorem 5 and Theorem 6, we set $\mathcal{R}_e = \mathcal{R}_w$. Additionally, we set $e_i + w_i \neq 64$ for $0 \leq i < m$ to prevent the emergence of iterative tails. Then we keep those candidates that ensure the relocation of the 10 bits in any two affected columns to at least 9 different columns. This guarantees that when F is equal to 0, the output of equal rows resulting from the active bits of E can be maximally distributed across different columns.

2. **Filter using experiments:** We record the candidates that move all bits to distinct columns for 11 in-kernel states with 6 active bits. This ensures that the column branch number for these states is 12. However, for 140 in-kernel states with 8 active bits (the next smallest number), we applied the same constraint but found that none of the parameters met the condition. Consequently, we relaxed the constraint to allow for the shifting of up to 2 active bits to a single column for in-kernel states with 8 active bits. Despite these filtering steps, numerous candidate sets remain. Therefore, we proceed by repeating the previous filters for the in-kernel states with 10 active bits and the states outside the kernel with few active bits. Finally, $\mathcal{R}_e = \mathcal{R}_w = \{0, 61, 49, 13, 19\}$ are determined.

Comparison with the parameter selection process of the TCPM in [HDRM23]. The parameter search process for both the TCPM and symmetric TCPM consists of two main stages. The first stage aims to reduce the total parameter space using equivalence classes and theorems related to the linear layer. The second stage involves selecting the best parameters by experimentally testing the performance of the remaining candidates. In the first stage, we use Theorem 1, 5, 6 and 9, which are specific to the symmetric TCPM. In the second stage, since the symmetric TCPM's differential branch histogram is the same as its linear branch histogram, we only need to filter candidate parameters based on differential performance. In contrast, the TCPM search process first filters based on differential performance and then applies linear filtering. Additionally, the differential screening conditions for the symmetric TCPM are considered more carefully. For instance, we filter candidates by minimizing the number of 6-bit states in the kernel, followed by minimizing the number of 8-bit states (the next smallest states), whereas the TCPM search only performs filtering based on minimizing the number of 6-bit states in the kernel.

7 Applications: Cryptographic Permutations using the Symmetric TCPM

In this section, we present the design of two 320-bit iterative cryptographic permutations, namely Gaston-S and SBD which use the symmetric TCPM as the underlying linear diffusion layer. We also compare their security and performance (in both hardware and software) with Gaston. Additionally, although the primary comparison is with Gaston, we also provide the security and performance of Ascon for reference.

7.1 Specifications of Gaston-S and SBD

Similar to Gaston, Gaston-S and SBD operate on 320-bit states with $m = 5$ rows and $n = 64$ columns and the round function consists of a linear layer followed by a non-linear layer. Gaston-S and SBD have the same linear layer, which uses the symmetric TCPM as the mixing layer and is preceded and followed by the row shift step ρ and a round constant addition. The main difference between Gaston-S and SBD is that the Gaston-S uses the χ-mapping that operates in parallel on the 5-bit columns as the non-linear layer and SBD uses a 5-bit S-box with degree 4. Algorithm 1 describes Gaston-S. The shift offset parameters and round constants of Gaston-S are specified in Table 3. The values of the round constants are the same as those used in Ascon and Gaston. The test vectors of Gaston-S and SBD are provided in Appendix G.

Algorithm 1: Definition of Gaston-S

```
Input: Number $q$ of rounds
1 for round index $i$ from $1-q$ up to 0 do $A = \mathrm{R}_i(A)$;
2 The round function R :
3 for $j$ from 0 to 4 do $A_j \leftarrow (A_j \lll e_j)$ ;
4 $E \leftarrow \left(\sum_{k=0}^{4} A_k + \left(\left(\sum_{k=0}^{4} A_k\right) \lll r\right)\right)$
5 $F \leftarrow \left(\sum_{k=0}^{4} (A_k \lll t_k) + \left(\sum_{k=0}^{4} (A_k \lll t_k) \lll s\right)\right)$
6 for $j$ from 0 to 4 do $A_j \leftarrow (A_j + (E + (F \lll t_j)) \lll u)$ ;
7 for $j$ from 0 to 4 do $A_j \leftarrow (A_j \lll e_j)$ ;
8 $A_0 \leftarrow A_0 + C_i$
9 for $j$ from 0 to 4 do $A_j \leftarrow A_j + \overline{A_{j+1}} \cdot A_{j+2}$ ;
```

Motivation for SBD Due to the usage of 2-degree S-boxes, the degree upper bounds of Gaston and Gaston-S grow slowly (see Table 5) which could lead to some potential cube-like attacks. Therefore, we aim to use a high-degree S-box in combination with the symmetric TCPM to resist such attacks. To achieve efficiency, we also want this S-box to be low-latency in hardware. Thus, we search for 5-bit S-boxes using NAND3 and NAND2 gates following Leander et al.'s approach [LMMR21] and a 4-degree S-box is selected. The

Table 3: Parameters for the linear diffusion layer of Gaston-S and the round constants C_i.

Index	r	s	t_k	u	e_j
0	1	36	6	26	0
1			10		61
2			27		49
3			41		13
4			50		19

i	C_i	i	C_i	i	C_i
-11	`0xF0`	-7	`0xB4`	-3	`0x78`
-10	`0xE1`	-6	`0xA5`	-2	`0x69`
-9	`0xD2`	-5	`0x96`	-1	`0x5A`
-8	`0xC3`	-4	`0x87`	0	`0x4B`

Table 4: Differential and linear bounds. The exact numbers in the table represent the minimum values.

Differential				
	Gaston-S	SBD	Gaston	Ascon
Column-wise branch number	12	12	12	4
Weight of 2-round trail	24	24	24	8
Weight of 3-round trail	≤ 148	≤ 120.32	≤ 106	40
Linear				
Column-wise branch number	12	12	4	4
Weight (C^2) of 2-round trail	24	24	8	8
Weight (C^2) of 3-round trail	≤ 141	≤ 81.46	34	28

code for searching this S-box is available in GitHub repository. Interestingly, this S-box is actually not new and has been found by Rasoolzadeh in [Ras22][2]. In Appendix C, we provide the truth table, DDT, LAT, and a bitslice implementation of this S-box.

7.2 Security Analysis of Gaston-S and SBD

In this section, we analyze the linear security, differential security, and integral security of Gaston-S and SBD.

7.2.1 Differential and Linear Trails Analysis

We search the differential and linear trails of Gaston-S and SBD using constraint programming. Similar to the method of searching for linear trails of Gaston used in Section 3.2, we set $E + F^{t_0} = 0, \ldots, E + F^{t_4} = 0$ at either the first or second round. The results of differential and linear trials of Gaston-S and SBD are shown in Table 4 while the actual trails are given in Appendices E and F.

From Table 4, we observe that the column-wise linear branch number of the mixing layer of Gaston-S is three times larger than that of Gaston. Although Gaston-S and Gaston have the same column-wise differential branch number, the best 3-round differential and linear trails of Gaston-S exhibit significantly higher weights compared to those of Gaston. One of the reasons is that the bit weight histogram of the symmetric TCPM used in Gaston-S has lower upper left tails than the bit weight histogram of the TCPM used in Gaston (see Appendix A.6). However, the best 3-round differential and linear trails of SBD exhibit lower weights compared to those of Gaston-S. This is mainly because, for the symmetric TCPM we used, the S-box in the SBD does not have a good distribution of DDT and LAT (although we could find S-boxes with better distributions, they were discarded due to their less efficient implementations). Nevertheless, SBD is sufficient to achieve good linear and differential security.

[2]Rasoolzadeh provided this S-box in the fourth line of the file `RESULTS/S5/S5D3C1_L16U6.txt` available at the following URL: `https://gitlab.science.ru.nl/shahramr/LowLatencySBoxes`.

Table 5: Degree upper bounds for Ascon, Gaston, Gaston-S and SBD.

Rounds	3	4	5	6	7	8	9	10	11
Gaston	5	11	18	35	65	130	258	300	315
Gaston-S	5	11	18	35	65	130	258	300	315
SBD	19	65	259	305	319	320	320	320	320
Ascon	5	11	18	35	65	130	258	300	315

Table 6: A comparison of round-based implementations of Ascon, Gaston, Gaston-S, and SBD.

NanGate 15nm			
	Time delay (ns)	Area (GE)	Power (mW)
Gaston	0.0565	1819	0.3861
Gaston-S	0.0572	2241	0.4436
SBD	0.0588	2340	0.4756
Ascon	0.0328	1714	0.3041
TSMC 28nm			
	Time delay (ns)	Area (GE)	Power (mW)
Gaston	0.1131	9936	0.025815
Gaston-S	0.1053	13152	0.036694
SBD	0.1016	13792	0.037487
Ascon	0.0631	9963	0.025513
NanGate 45nm			
	Time delay (ns)	Area (GE)	Power (mW)
Gaston	0.6160	6706	0.1324
Gaston-S	0.6060	8004	0.1612
SBD	0.6452	8557	0.1739
Ascon	0.3727	6803	0.1395

7.2.2 Integral Analysis

Using the division property [Tod15], we present the degree upper bounds for Gaston-S, SBD, and Gaston in Table 5. For Gaston-S and Gaston, up to 7 rounds can be distinguished with less than 2^{128} data. For SBD, up to 5 rounds can be distinguished with less than 2^{128} data.

7.3 Implementation Efficiency of Gaston-S and SBD

In this section, we evaluate the hardware and software performances of Gaston-S and SBD.

7.3.1 Hardware Performance

We implemented Gaston, Gaston-S, and SBD in VHDL, synthesized it with the Synopsys Design Compiler S-2021.06-SP3, and repeated the analysis with 3 different standard cell libraries, 1 of which is manufacturable cell libraries from a commercial foundry (TSMC 28nm), while the remaining 2 are open-source libraries which are not manufacturable (Nangate 15/45nm) but can be used for producing universally comparable and reproducible synthesis results. The results are shown in Table 6.

From Table 6, we observe that Gaston, Gaston-S, and SBD have similar implementation speeds. This is due to their linear layers sharing the same maximum XOR depth: $\lceil \log_2(4m+1) \rceil$ and the S-box of SBD has similar implementation efficiency comparable to that of the χ-mapping. Gaston-S and SBD require more area and power than Gaston because of the additional row cyclic shifts in the symmetric TCPM.

Table 7: Software performance in cycles per round.

	Gaston	Gaston-S	SBD	Ascon
Intel Xeon Gold 6348 CPU(x86_64)	16.04	18.21	24.62	13.31
Cortex-A76(ARMv8)	16.02	16.82	23.72	12.65

7.3.2 Software Performance

We evaluated the efficiency of Gaston, Gaston-S, and SBD algorithms on both the x86_64 and ARMv8 architectures. The results are presented in Table 7.

On x86_64 architecture, Gaston-S is slightly slower than Gaston. However, on the ARMv8 architecture, the speed of Gaston-S closely matches that of Gaston due to the efficiency of 64-bit row cyclic shifts on ARMv8. In contrast, SBD's software implementation is significantly slower than both Gaston-S and Gaston, primarily because the software implementation of its S-box in SBD is slower.

8 Conclusions

In this paper, we presented deeper insights into the linear security of the TCPM by giving a theoretical proof of its linear branch number and showing that linear trails of Gaston have worse bounds than Ascon for 4 or more rounds. We then proposed the modified TCPM which is a generalization of the TCPM and that achieves both the differential branch number and linear branch number 12. Furthermore, we introduced a specialized form of the modified TCMP called the symmetric TCMP whose differential and linear branch histograms are the same. We then provided the concrete parameter set of the shift offsets of the symmetric TCMP for a state size of 320 bits. Employing this chosen symmetric TCMP as the linear diffusion layer, we proposed two new 320-bit iterative cryptographic permutations, namely Gaston-S and SBD.

We evaluated the security of Gaston-S and SBD against traditional attacks, and showed that the best 3 rounds differential and linear trails of these ciphers have much higher weights compared to Gaston. Moreover, we implemented Gaston-S and SBD in different hardware libraries and software platforms. Our implementation results indicated that both Gaston-S and SBD have reasonable performance compared to Gaston.

We believe that permutations like Gaston-S can be very competitive in applications that require high differential and linear security while SBD is competitive if higher security against algebraic attacks is additionally required. On another note, since the constructions of TCPMs, modified TCPMs, and symmetric TCPMs are generic, it is an interesting research direction to construct new cryptographic permutations (find suitable parameter offsets) with other state sizes.

Acknowledgment. The authors would like to thank the anonymous shepherd and other anonymous reviewers who have helped us improve the quality of this paper. This research is supported by the National Key Research and Development Program of China (Grant No. 2018YFA0704700), the National Natural Science Foundation of China (Grant Nos. 62032014, U2336207, 62072270), Department of Science & Technology of Shandong Province (Grant No. SYS202201), NationalKey R&D Program of China (Grant No. 2023YFA1009500), Quan Cheng Laboratory (Grant Nos. QCLZD202301, QCLZD202306).

References

[BDPVA09] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. Keccak specifications. *Submission to nist (round 2)*, 3(30):320–337, 2009.

[BDPVA13] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. Keccak. In *Annual international conference on the theory and applications of cryptographic techniques*, pages 313–314. Springer, 2013.

[BR$^+$00] PSLM Barreto, Vincent Rijmen, et al. The Whirlpool hashing function. In *First open NESSIE Workshop, Leuven, Belgium*, volume 13, page 14. Citeseer, 2000.

[BS91] Eli Biham and Adi Shamir. Differential cryptanalysis of DES-like cryptosystems. *Journal of CRYPTOLOGY*, 4:3–72, 1991.

[CTG16] Beierle Christof, Kranz Thorsten, and Leander Gregor. Lightweight multiplication in gf (2n) with applications to mds matrices; crypto 2016. lncs 9814, 2016.

[Dae95] Joan Daemen. *Cipher and hash function design strategies based on linear and differential cryptanalysis*. PhD thesis, Doctoral Dissertation, March 1995, KU Leuven, 1995.

[DEM15] Christoph Dobraunig, Maria Eichlseder, and Florian Mendel. Heuristic tool for linear cryptanalysis with applications to CAESAR candidates. In Tetsu Iwata and Jung Hee Cheon, editors, *Advances in Cryptology - ASIACRYPT 2015 - 21st International Conference on the Theory and Application of Cryptology and Information Security, Auckland, New Zealand, November 29 - December 3, 2015, Proceedings, Part II*, volume 9453 of *Lecture Notes in Computer Science*, pages 490–509. Springer, 2015.

[DEMS15] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Martin Schläffer. Cryptanalysis of ascon. In Kaisa Nyberg, editor, *Topics in Cryptology - CT-RSA 2015, The Cryptographer's Track at the RSA Conference 2015, San Francisco, CA, USA, April 20-24, 2015. Proceedings*, volume 9048 of *Lecture Notes in Computer Science*, pages 371–387. Springer, 2015.

[DEMS21] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Martin Schläffer. Ascon v1. 2: Lightweight authenticated encryption and hashing. *Journal of Cryptology*, 34:1–42, 2021.

[DHVAVK18] Joan Daemen, Seth Hoffert, Gilles Van Assche, and Ronny Van Keer. The design of Xoodoo and Xoofff. *IACR Transactions on Symmetric Cryptology*, pages 1–38, 2018.

[DL18] Sébastien Duval and Gaëtan Leurent. MDS matrices with lightweight circuits. *IACR Transactions on Symmetric Cryptology*, 2018(2):48–78, 2018.

[EME22] Johannes Erlacher, Florian Mendel, and Maria Eichlseder. Bounds for the security of ascon against differential and linear cryptanalysis. *IACR Trans. Symmetric Cryptol.*, 2022(1):64–87, 2022.

[GPT21] David Gérault, Thomas Peyrin, and Quan Quan Tan. Exploring differential-based distinguishers and forgeries for ASCON. *IACR Trans. Symmetric Cryptol.*, 2021(3):102–136, 2021.

[HDRM23] Solane El Hirch, Joan Daemen, Raghvendra Rohit, and Rusydi H. Makarim. Twin column parity mixers and Gaston - A new mixing layer and permutation. *IACR Cryptol. ePrint Arch.*, page 799, 2023.

[JV02] Daemen Joan and Rijmen Vincent. The design of Rijndael: AES-the advanced encryption standard. *Information Security and Cryptography*, 2002.

[KLSW17] Thorsten Kranz, Gregor Leander, Ko Stoffelen, and Friedrich Wiemer. Shorter linear straight-line programs for MDS matrices. *IACR Transactions on Symmetric Cryptology*, pages 188–211, 2017.

[LMM91] Xuejia Lai, James L. Massey, and Sean Murphy. Markov ciphers and differential cryptanalysis. In Donald W. Davies, editor, *Advances in Cryptology - EUROCRYPT '91, Workshop on the Theory and Application of of Cryptographic Techniques, Brighton, UK, April 8-11, 1991, Proceedings*, volume 547 of *Lecture Notes in Computer Science*, pages 17–38. Springer, 1991.

[LMMR21] Gregor Leander, Thorben Moos, Amir Moradi, and Shahram Rasoolzadeh. The SPEEDY family of block ciphers engineering an ultra low-latency cipher from gate level for secure processor architectures. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2021(4):510–545, 2021.

[LS16] Meicheng Liu and Siang Meng Sim. Lightweight MDS generalized circulant matrices. In *International Conference on Fast Software Encryption*, pages 101–120. Springer, 2016.

[LSS$^+$19] Shun Li, Siwei Sun, Danping Shi, Chaoyun Li, and Lei Hu. Lightweight iterative MDS matrices: how small can we go? *Cryptology ePrint Archive*, 2019.

[LW17] Chaoyun Li and Qingju Wang. Design of lightweight linear diffusion layers from near-MDS matrices. *Cryptology ePrint Archive*, 2017.

[MFW$^+$17] Kerry McKay, Larry Feldman, Gregory Witte, et al. Toward standardizing lightweight cryptography. Technical report, National Institute of Standards and Technology, 2017.

[MR22] Rusydi H. Makarim and Raghvendra Rohit. Towards tight differential bounds of ascon A hybrid usage of SMT and MILP. *IACR Trans. Symmetric Cryptol.*, 2022(3):303–340, 2022.

[Nat19] National Institute of Standards and Technology. Lightweight Cryptography (LWC) Standardization project, 2019. `https://csrc.nist.gov/projects/lightweight-cryptography`.

[Ras22] Shahram Rasoolzadeh. Low-latency boolean functions and bijective s-boxes. *IACR Trans. Symmetric Cryptol.*, 2022(3):403–447, 2022.

[RDP$^+$96] Vincent Rijmen, Joan Daemen, Bart Preneel, Antoon Bosselaers, and Erik De Win. The cipher SHARK. In *Fast Software Encryption: Third International Workshop Cambridge, UK, February 21–23 1996 Proceedings 3*, pages 99–111. Springer, 1996.

[SD18] Ko Stoffelen and Joan Daemen. Column parity mixers. 2018.

[SKOP15] Siang Meng Sim, Khoongming Khoo, Frédérique Oggier, and Thomas Peyrin. Lightweight MDS involution matrices. In *Fast Software Encryption: 22nd International Workshop, FSE 2015, Istanbul, Turkey, March 8-11, 2015, Revised Selected Papers 22*, pages 471–493. Springer, 2015.

[Tod15] Yosuke Todo. Structural evaluation by generalized integral property. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 287–314. Springer, 2015.

[VKS22] Ayineedi Venkateswarlu, Abhishek Kesarwani, and Sumanta Sarkar. On the lower bound of cost of MDS matrices. *IACR Transactions on Symmetric Cryptology*, pages 266–290, 2022.

Appendix

A The Symmetric Twin Column Parity Mixers

A.1 Proof of Theorem 2

Proof. According to Eqn. (9), $\boldsymbol{L}^T$ applied to a two-dimensional state can be expressed as

$$\begin{aligned}
B_i &= A_i + \sum_{k=0}^{m-1}(A_k \ggg u) + \sum_{k=0}^{m-1}(A_k \ggg (u+r)) \\
&+ \sum_{k=0}^{m-1}(A_k \ggg (u+t_k+t_i)) + \sum_{k=0}^{m-1}(A_k \ggg (u+t_k+t_i+s)) \\
&= A_i + \sum_{k=0}^{m-1}(A_k \lll (n-u)) + \sum_{k=0}^{m-1}(A_k \lll (n-(u+r))) \\
&+ \sum_{k=0}^{m-1}(A_k \lll (n-(u+t_k+t_i))) + \sum_{k=0}^{m-1}(A_k \lll (n-(u+t_k+t_i+s))) \\
&= A_i + \sum_{k=0}^{m-1}(A_k \ggg u') + \sum_{k=0}^{m-1}(A_k \ggg (u'+r')) \\
&+ \sum_{k=0}^{m-1}(A_k \ggg (u'+t'_k+t'_i)) + \sum_{k=0}^{m-1}(A_k \ggg (u'+t'_k+t'_i+s')),
\end{aligned}$$

where $u' = n-u, r' = n-r, s' = n-s,$ and $t'_k = n-t_k$ for $0 \le k < m$. □

A.2 Proof of Theorem 4

Proof. Assume $\boldsymbol{L}$ is an $m \times n$ dimensional symmetric TCPM. Let $H_{c,\boldsymbol{L}}$ and $H_{c,\boldsymbol{L}^T}$ be bit differential branch histogram and bit linear branch histogram of $\boldsymbol{L}$, respectively. Now, if $H_{c,\boldsymbol{L}}(w) = x$, then there are x differential trails

$$\boldsymbol{D^0} \xrightarrow{\boldsymbol{L}} \boldsymbol{O^0}, \cdots, \boldsymbol{D^{x-1}} \xrightarrow{\boldsymbol{L}} \boldsymbol{O^{x-1}}$$

satisfying

$$wt(\boldsymbol{D^0}) + wt(\boldsymbol{O^0}) = w, \ldots, wt(\boldsymbol{D^{x-1}}) + wt(\boldsymbol{O^{x-1}}) = w.$$

By Theorem 3, there exist x linear trails

$$\boldsymbol{\alpha^0} \xrightarrow{\boldsymbol{L}} \boldsymbol{\beta^0}, \ldots, \boldsymbol{\alpha^{x-1}} \xrightarrow{\boldsymbol{L}} \boldsymbol{\beta^{x-1}}$$

and they satisfy

$$\begin{cases}
wt(\boldsymbol{\beta^0}) = wt(\boldsymbol{D^0}), wt(\boldsymbol{\alpha^0}) = wt(\boldsymbol{O^0}) \\
wt(\boldsymbol{\beta^1}) = wt(\boldsymbol{D^1}), wt(\boldsymbol{\alpha^1}) = wt(\boldsymbol{O^1}) \\
\qquad\qquad \cdots \\
wt(\boldsymbol{\beta^{x-1}}) = wt(\boldsymbol{D^{x-1}}),\ wt(\boldsymbol{\alpha^{x-1}}) = wt(\boldsymbol{O^{x-1}})
\end{cases}$$

Hence,

$$wt(\boldsymbol{\alpha^0}) + wt(\boldsymbol{\beta^0}) = w, \ldots, wt(\boldsymbol{\alpha^{x-1}}) + wt(\boldsymbol{\beta^{x-1}}) = w.$$

That is $H_{c,\boldsymbol{L}^T}(w) \ge x$, then $H_{c,\boldsymbol{L}^T}(w) \ge H_{c,\boldsymbol{L}}(w)$. Similarly, assuming that $H_{c,\boldsymbol{L}^T} = y$, we can deduce that $H_{c,\boldsymbol{L}}(w) \ge H_{c,\boldsymbol{L}^T}(w)$. Since $H_{c,\boldsymbol{L}}(w) = H_{c,\boldsymbol{L}^T}(w)$, the proof is complete. □

A.3 Proof of Theorem 5

Proof. Assume $\boldsymbol{\alpha} \xrightarrow{\rho} \boldsymbol{\alpha'} \xrightarrow{\boldsymbol{L}} \boldsymbol{\beta'} \xrightarrow{\rho} \boldsymbol{\beta}$ and $\boldsymbol{D} \xrightarrow{\rho} \boldsymbol{D'} \xrightarrow{\boldsymbol{L}} \boldsymbol{O'} \xrightarrow{\rho} \boldsymbol{O}$. By Theorem 3, we have

$$\beta'_{i,n-j} = D'_{i,j} \text{ and } \alpha'_{i,n-j} = O'_{i,j}, \text{ for } 0 \leq i < m, 0 \leq j < n.$$

From the expression for ρ, we have

$$D_{i,j} = D'_{i,(j-\omega_i) \bmod n},\ O_{i,j} = O'_{i,(j+\omega_i) \bmod n},$$

$$\alpha_{i,j} = \alpha'_{i,(j-\omega_i) \bmod n},\ \beta_{i,j} = \beta'_{i,(j+\omega_i) \bmod n}.$$

Therefore,

$$D_{i,j} = D'_{i,(j-\omega_i) \bmod n} = \beta'_{i,(n-(j-\omega_i)) \bmod n} = \beta_{i,(n-(j-\omega_i)-\omega_i) \bmod n} = \beta_{i,n-j},$$

$$O_{i,j} = O'_{i,(j-\omega_i) \bmod n} = \alpha'_{i,(n-(j-\omega_i)) \bmod n} = \alpha_{i,(n-(j-\omega_i)-\omega_i) \bmod n} = \alpha_{i,n-j}.$$

□

A.4 Proof of Theorem 9

Proof. States with only one active bit in row a and column b have $E = X^b + X^{b+r}$ and $F = X^{b+t_a} + X^{b+s+t_a}$. So, $E + X^{t_i}F = X^b + X^{b+r} + X^{b+t_a+t_i} + X^{b+t_a+t_i+s}$, where $0 \leq i < m$. Since $r, s \neq 0$, two bits that are indexed by b and $b+r$ in E are active. Also, any two bits that are indexed by $b + t_a + t_i$ and $b + t_a + t_i + s$ in $X^{t_i}F$ are active.

The indexes of active bits that appear in $X^{t_i}F$ are $\{b + t_a + t_0, \dots, b + t_a + t_{m-1}\}$ and $\{b+t_a+t_0+s, \dots, b+t_a+t_{m-1}+s\}$. Note that the elements in $\{t_a+t_0, t_a+t_1, \dots, t_a+t_{m-1}\}$ are pairwise unequal. This is also the case for elements in $\{b+t_a+t_0, \dots, b+t_a+t_{m-1}\}$ or $\{b+t_a+t_0, \dots, b+t_a+t_{m-1}\}$. Therefore, the indexes of active bits in E collides with at most two indexes in $\{b+t_a+t_0, \dots, b+t_a+t_{m-1}\}$ or two indexes in $\{b+t_a+t_0, \dots, b+t_a+t_{m-1}\}$. Therefore, the output active bits are at least $4m - 4$ and the branch number is at least $4m - 3$. □

A.5 The Histogram for $\mathcal{R}_\theta$

Table 8: 2-dimensional bit weight histogram for $\mathcal{R}_\theta = (26, 1, 36, 6, 10, 27, 41, 50)$. The row and column indices represent the number of active input bits and active output bits of the mixing layer, respectively.

	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
1																5		
2															1		20	
3														2		20		54
4									7						1		38	
5														19		91		471
6	11								4		190		7		11		321	
7												1		66		628		2921
8			140						56		176		2587		348		845	
9												3		166		2598		19168
10					1653		17		7		2491		4642		44098		13850	

A.6 The Histogram of the Difference

Table 9: The histogram of the difference between histograms $H_{b,L_1}(w)$ and $H_{b,L_2}(w)$, L_1 is the symmetric TCPM used in Gaston-S and L_2 is the TCPM used in Gaston, i.e., $\#\{\boldsymbol{D} \in \mathcal{D} \text{ with } \mathcal{B}_{c,L_1}(\boldsymbol{D}) = w\} - \#\{\boldsymbol{D} \in \mathcal{D} \text{ with } \mathcal{B}_{c,L_2}(\boldsymbol{D}) = w\}$. The blue value represents the difference is negative and the red value represents the difference is positive.

	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
1																0		
2															+1		-9	
3								-1						+2		+15		-95
4									-17				-1		-13		-29	
5										-158		-3		+2		-198		-35
6	0						-18		-11		-563		-35		-121		-2691	
7				-1				-200		-143		-4446		-512		-1302		-26646
8			-206		-18		-25		-1365		-1256		-23262		-6771		-23329	
9						-94		-303		-12900		-16425		-206896		-78901		-264855
10					-9299		-1240		-3132		-92015		-136526		-1269739		-880659	

B Lemmas from [HDRM23]

Lemma 1 ([HDRM23])**.** *For even n, let $\{i, j\} \subset \{0, 1, \dots, m-1\}$ and $t_i = t_j + n/2$, then the state $\boldsymbol{D}$ with two active rows $D_i = D_j = 1 + X^{n/2}$ is in the kernel.*

Proof. The state $\boldsymbol{D}$ has $P = \left(1 + X^{n/2}\right) + \left(1 + X^{n/2}\right) = 0$ and $Q = X^{t_j}\left(1 + X^{n/2}\right) + \left(X^{t_j + n/2}\right)\left(1 + X^{n/2}\right) = 0$. Since it is in the kernel. □

Lemma 2 ([HDRM23])**.** *Let $\{i, j, k\} \subset \{0, 1, \dots, m-1\}$ and $t_i + t_j = 2t_k$, then the state $\boldsymbol{D}$ with three active rows $D_i = X^{t_k}, D_j = X^{t_i}$ and $D_k = X^{t_k} + X^{t_i}$ is in the kernel.*

Proof. The state $\boldsymbol{D}$ has $P = X^{t_k} + X^{t_i} + X^{t_k} + X^{t_i} = 0$ and $Q = X^{t_i + t_k} + X^{t_j + t_i} + X^{t_k + t_k} + X^{t_k + t_i} = 0$. Since it is in the kernel. □

Lemma 3 ([HDRM23])**.** *Let $\{i, j, k, l\} \subset \{0, 1, \dots, m-1\}$ and $t_i + t_j = t_k + t_l$, then the state with 4 active rows $D_i = 1, D_j = X^{t_k - t_j}, D_k = 1$ and $D_l = X^{t_i - t_l}$ is in the kernel.*

Proof. The state $\boldsymbol{D}$ has $1 + X^{t_k - t_j} + 1 + X^{t_i - t_l} = 0$ and $D_i = X^{t_i} + X^{t_k} + X^{t_k} + X^{t_i} = 0$. Since it is in the kernel. □

To avoid in-kernel states with 4 active bits (or 2 active bits). For any set $\{i, j\} \subset \{0, 1, \dots, m-1\}$ (for even n):

$$2t_i \neq 2t_j.$$

For any sets $\{i, j\} \subset \{0, 1, \dots, m-1\}$ and $k \in \{0, 1, \dots, m-1\}$ with $k \notin \{i, j\}$:

$$t_i + t_j \neq 2t_k.$$

For any two sets $\{i, j\} \subset \{0, 1, \dots, m-1\}$ and $\{k, l\} \subset \{0, 1, \dots, m-1\}$ and $\{i, j\} \cap \{k, l\} = \varnothing$:

$$t_i + t_j \neq t_k + t_l.$$

C The degree-4 S-box

Algorithm 2: Bitsliced implementation of the degree-4 S-box

```
Input: $x_0, x_1, x_2, x_3, x_4$
// step 1
$n_0 \leftarrow INV(x_0)$
$n_1 \leftarrow INV(x_1)$
$n_2 \leftarrow INV(x_2)$
$n_3 \leftarrow INV(x_3)$
$n_4 \leftarrow INV(x_4)$
// step 2
$t_0 \leftarrow NAND2(x_0, x_1)$
$t_1 \leftarrow NAND2(x_2, x_3)$
$t_2 \leftarrow NAND3(x_0, n_2, x_4)$
$t_3 \leftarrow NAND2(x_4, n_0)$
$t_4 \leftarrow NAND2(x_1, n_2)$
$t_5 \leftarrow NAND3(x_4, n_1, x_3)$
$t_6 \leftarrow NAND2(x_3, n_4)$
$t_7 \leftarrow NAND2(n_0, n_1)$
$t_8 \leftarrow NAND3(x_3, x_0, n_2)$
$t_9 \leftarrow NAND2(n_2, n_3)$
$t_{10} \leftarrow NAND2(n_4, x_0)$
$t_{11} \leftarrow NAND3(n_2, x_4, n_1)$
$t_{12} \leftarrow NAND2(n_1, x_2)$
$t_{13} \leftarrow NAND2(n_3, x_4)$
$t_{14} \leftarrow NAND3(n_1, x_3, x_0)$
// step 3
$y_0 \leftarrow NAND3(t_0, t_1, t_2)$
$y_1 \leftarrow NAND3(t_3, t_4, t_5)$
$y_2 \leftarrow NAND3(t_6, t_7, t_8)$
$y_3 \leftarrow NAND3(t_9, t_{10}, t_{11})$
$y_4 \leftarrow NAND3(t_{12}, t_{13}, t_{14})$
Output: $y_0, y_1, y_2, y_3, y_4$
```

Table 10: The truth table of the degree-4 S-box

x	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f	10	11	12	13	14	15	16	17	18	19	1a	1b	1c	1d	1e	1f
$\mathcal{S}(x)$	6	f	4	e	5	d	15	1d	a	b	c	8	0	9	14	18	2	13	7	1f	3	1	17	19	1a	1b	1e	1c	12	11	16	10

Table 11: Differential and linear profile of the degree-4 S-box.

(a) Differential distribution table: $\mathrm{DDT}[\alpha, \beta] = |\{x : \mathcal{S}(x \oplus \alpha) \oplus \mathcal{S}(x) = \beta\}|$

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f	10	11	12	13	14	15	16	17	18	19	1a	1b	1c	1d	1e	1f
0	32	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
1	.	4	4	2	2	.	2	.	4	4	2	.	2	.	2	.	.	2	.	.	.	.	.	.	2	.	.	.	.	.	.	.
2	.	4	2	2	4	2	2	2	.	.	.	.	2	.	.	.	4	2	.	.	4	.	.	.	2	.	.	.	.	.	.	.
3	.	.	4	.	.	2	2	4	2	.	.	2	.	.	.	.	.	.	.	.	2	.	2	.	6	.	2	.	.	4	.	.
4	.	2	4	2	.	.	2	.	4	.	4	.	2	.	.	.	4	2	2	2	.	.	.	.	2	.	.	.	.	.	.	.
5	.	.	.	4	.	.	.	.	2	2	4	6	.	.	2	.	2	.	.	.	2	.	.	.	.	2	.	2	2	.	2	.
6	.	4	.	2	2	.	.	.	.	.	2	2	6	2	.	.	.	.	2	4	.	2	.	.	.	.	.	.	.	.	4	.
7	.	.	.	.	2	2	2	.	2	2	2	2	.	2	2	2	.	.	2	.	.	.	.	.	.	.	2	4	2	.	.	2
8	.	4	.	2	4	4	2	.	4	2	.	.	2	.	.	.	2	2	.	.	.	.	.	.	2	2	.	.	.	.	.	.
9	.	2	2	.	.	2	.	2	.	4	.	.	4	6	.	2	.	.	2	2	.	.	.	.	.	2	.	2	.	.	.	.
a	.	.	.	.	2	4	.	2	2	.	2	.	.	.	2	2	.	4	.	.	2	6	.	.	.	.	.	.	2	2	.	.
b	.	.	.	2	2	2	2	.	.	.	.	2	.	2	2	2	.	.	.	2	.	.	.	2	.	2	.	.	4	4	2	.
c	.	.	2	.	.	2	6	.	.	2	.	.	.	.	.	4	4	2	.	.	.	2	2	.	.	4	2	.	.	.	.	.
d	.	2	.	.	.	.	.	4	.	2	.	2	.	.	2	4	.	2	.	2	.	.	.	2	2	.	2	2	2	2	.	.
e	.	.	2	2	2	2	.	2	.	2	.	.	.	2	2	.	.	.	2	.	2	2	2	2	.	.	.	.	.	4	.	2
f	.	2	.	2	.	2	.	.	.	.	.	.	2	6	2	.	.	.	2	.	.	.	2	2	.	.	.	2	.	.	4	4
10	.	.	4	2	4	.	2	.	2	.	.	.	2	.	.	.	4	2	4	.	2	.	.	.	2	.	.	.	2	.	.	.
11	.	.	.	.	4	.	.	.	.	2	2	.	2	2	4	.	2	6	.	.	.	2	2	.	.	.	.	4	.	.	.	.
12	.	2	.	.	.	.	4	.	.	2	.	.	.	.	.	.	2	.	2	2	4	.	6	2	.	2	.	.	2	2	.	.
13	.	.	.	.	.	2	.	.	2	.	2	.	2	.	.	.	2	.	2	.	2	2	2	4	.	2	2	.	2	.	2	2
14	.	.	2	.	2	2	.	2	.	.	2	.	.	.	2	.	.	.	4	2	.	.	.	2	4	.	6	.	.	.	2	.
15	.	.	.	.	.	.	2	2	2	.	2	.	2	.	.	2	.	.	.	2	.	.	2	.	.	4	2	4	2	2	2	.
16	.	.	2	2	.	.	.	2	.	.	.	.	.	.	4	2	.	2	2	.	.	2	2	2	.	2	.	2	2	.	4	.
17	.	.	2	.	.	2	2	.	.	.	2	.	.	2	.	2	.	.	.	.	.	.	.	2	2	.	6	.	2	4	.	4
18	.	2	.	6	.	.	.	.	4	.	.	2	.	2	.	.	.	.	2	.	2	.	.	4	2	.	2	.	4	.	.	.
19	.	2	.	.	.	2	.	.	.	2	2	2	.	2	.	4	2	.	.	2	2	2	.	.	2	2	.	.	.	2	.	2
1a	.	.	.	.	.	.	.	2	.	.	.	.	2	2	2	.	2	.	.	4	2	2	.	4	2	2	.	2	.	2	2	.
1b	.	.	.	.	2	.	.	.	.	.	2	.	2	.	.	2	.	2	.	.	2	6	.	.	.	2	2	4	.	.	2	4
1c	.	2	2	.	.	.	.	2	.	2	2	2	.	.	.	.	.	2	2	2	2	.	2	.	.	.	2	2	.	.	4	2
1d	.	.	.	2	.	.	.	.	2	2	.	6	.	.	.	.	.	.	2	2	2	.	4	2	.	.	.	.	.	2	2	4
1e	.	.	.	.	.	.	2	2	.	2	.	2	.	.	.	4	2	2	2	.	.	.	6	.	.	.	.	2	.	2	.	4
1f	.	.	.	.	.	.	.	4	.	.	.	2	.	2	4	.	.	.	.	4	.	2	2	.	.	4	2	.	4	.	.	2

(b) Linear approximation table: $\mathrm{LAT}[\alpha, \beta] = \left|\left\{x : \alpha^\top \cdot x = \beta^\top \cdot \mathcal{S}(x)\right\}\right| - 16$

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f	10	11	12	13	14	15	16	17	18	19	1a	1b	1c	1d	1e	1f
0	16	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
1	.	6	-4	-2	-4	2	-4	-2	8	-2	.	2	-4	2	.	2	2	.	2	4	2	.	-2	.	2	.	-2	.	2	.	2	4
2	.	-4	-4	-4	8	.	-4	.	2	2	2	-2	2	-2	2	2	6	-2	2	-2	-2	2	2	2	.	4	.	.	.	.	.	4
3	.	6	.	-2	4	6	.	2	-2	.	-2	.	2	.	-2	4	.	2	-4	-2	.	-2	.	6	6	-4	2	.	-2	.	-2	.
4	.	4	-8	4	-2	-2	-2	-2	-6	-2	2	-2	.	.	.	.	4	4	.	.	-2	2	2	-2	2	2	-2	-2	-4	.	.	-4
5	.	2	4	-6	2	.	2	-4	-2	.	-2	4	.	-2	4	-2	2	.	-2	.	-4	-2	-4	-6	4	2	-4	-2	-2	.	2	.
6	.	.	-4	.	2	2	-2	2	.	4	.	.	-6	-2	2	2	-6	2	-6	-2	.	.	.	-4	-2	2	2	-6	4	.	.	.
7	.	2	.	-6	-2	.	2	-4	.	-2	.	-2	-2	-4	2	.	.	2	.	-6	6	.	2	4	-4	2	-4	2	2	.	-2	-4
8	.	-8	-2	-2	-6	2	.	.	4	.	-2	2	2	-2	-4	.	4	4	-2	-2	-2	-2	.	.	4	.	2	-2	2	-2	.	-4
9	.	-2	2	4	2	.	.	2	4	-2	-2	-4	-6	4	.	-2	2	.	-4	-2	.	-2	-2	.	2	4	-4	2	-4	-2	-6	.
a	.	4	2	2	-2	-2	.	4	2	-2	-4	-4	4	-4	-2	2	2	-6	.	-4	.	4	-2	-2	.	.	-2	-6	2	-2	.	.
b	.	-2	-2	-4	-2	-8	.	2	-2	4	.	-2	.	2	-2	.	.	2	-2	.	2	.	-4	2	2	-4	-4	-2	.	6	-2	4
c	.	4	-2	2	.	.	6	-2	6	6	.	.	2	-2	-4	.	.	.	-2	-2	-4	.	2	-2	-2	2	.	4	-2	6	.	.
d	.	2	2	.	.	-2	-2	.	2	8	-4	-2	-2	.	4	-6	2	.	.	2	2	4	4	2	4	-2	2	.	.	-2	2	-4
e	.	.	2	-2	.	.	2	-2	.	.	-6	-2	4	4	-2	2	-2	6	.	4	2	2	4	.	-2	6	.	-4	-2	-2	.	4
f	.	2	-2	.	.	2	2	4	.	2	2	-4	4	6	2	-4	.	2	2	-4	.	-6	-2	.	.	2	-2	.	4	-2	6	.
10	.	2	6	.	-4	2	-2	4	-4	2	2	.	-4	-2	-2	.	8	2	-2	.	.	-2	2	.	-4	2	2	.	.	2	2	4
11	.	.	6	2	.	-4	-2	-2	4	.	6	-2	.	.	2	6	-2	6	.	-4	-2	2	.	.	2	-2	.	.	-2	-2	4	.
12	.	-2	-2	.	-4	2	6	.	-2	4	.	2	-2	4	4	6	2	-4	.	-2	2	4	-4	2	.	2	2	.	-4	-2	2	.
13	.	.	-2	-2	.	.	6	6	2	-6	.	.	-2	-2	4	-4	.	4	2	-2	.	4	2	-2	2	-2	4	.	-2	2	.	4
14	.	2	-2	.	2	-4	4	-2	2	.	.	-2	.	-2	2	.	4	2	-2	4	-2	-4	-4	2	-6	-4	4	-2	.	-6	-2	.
15	.	.	-2	2	-2	-2	-4	.	-2	2	-8	.	.	-4	2	2	-2	2	4	-4	.	-4	-2	-2	.	.	2	6	-2	-2	.	4
16	.	-2	-2	.	-2	.	.	-2	.	-2	2	-4	2	-4	.	-2	-2	-4	-8	2	4	-2	2	.	2	.	.	2	-4	-2	6	4
17	.	.	-2	-2	2	-2	.	4	.	.	-2	6	-2	2	-4	.	.	.	-2	-2	-2	2	4	.	-4	-4	-6	2	-2	-6	4	.
18	.	2	.	-6	-6	.	-2	4	.	2	4	-2	2	.	2	.	-4	-2	.	2	-6	.	2	.	.	2	.	2	-2	-4	-6	.
19	.	.	.	-4	2	-2	2	2	.	.	.	-4	-6	-2	-6	2	-2	-2	6	2	.	-4	.	.	2	2	2	-2	-4	.	4	-4
1a	.	-2	.	2	-2	4	2	-4	-2	.	-2	-4	-4	-2	.	-2	-2	.	2	.	-8	2	.	6	.	-2	-4	-2	2	.	2	4
1b	.	.	.	.	-2	-2	-2	-2	2	-2	-2	2	.	4	4	.	.	-4	.	-4	-2	-6	6	2	-2	-2	2	-6	-4	4	.	.
1c	.	2	.	2	.	-6	4	-2	-2	.	2	4	-2	.	-2	.	.	-2	.	-2	.	-2	4	2	6	4	2	.	6	-4	-2	4
1d	.	.	.	4	.	.	.	4	2	2	2	6	2	-6	2	-2	-2	2	2	2	2	-2	-2	6	.	4	-4	-4	-4	.	.	.
1e	.	-2	.	2	.	2	4	2	.	2	.	-2	.	-2	4	6	2	.	2	4	2	-4	6	-4	2	-4	-6	.	2	.	-2	.
1f	.	.	.	.	.	-4	.	4	.	-4	-4	.	.	.	4	4	.	.	-4	4	-4	.	.	4	.	4	.	4	4	4	4	-4

D Gaston

D.1 Specification of Gaston

Algorithm 3: Definition of Gaston

```
  Input: Number $q$ of rounds
1 for round index $i$ from $1 - q$ up to 0 do $A = \mathrm{R}_i(A)$;
2 The round function R :
3 for $j$ from 0 to 4 do $A_j \leftarrow (A_j \lll e_j)$ ;
4 $E \leftarrow \left(\sum_{k=0}^{4} A_k + \left(\sum_{k=0}^{4} A_k \lll r\right)\right)$
5 $F \leftarrow \left(\sum_{k=0}^{4} (A_k \lll t_k) + \left(\sum_{k=0}^{4} (A_k \lll t_k) \lll s\right)\right)$
6 for $j$ from 0 to 4 do $A_j \leftarrow (A_j + (E + F) \lll u)$ ;
7 for $j$ from 0 to 4 do $A_j \leftarrow (A_j \lll w_j)$ ;
8 $A_0 \leftarrow A_0 + C_i$
9 for $j$ from 0 to 4 do $A_j \leftarrow A_j + \overline{A_{j+1}} \cdot A_{j+2}$ ;
```

Table 12: Parameters for the linear diffusion layer of Gaston and the round constants C_i.

Index	r	s	t_k	u	w_j	e_j
0	1	18	25	23	0	0
1			32		56	60
2			52		31	22
3			60		46	27
4			63		43	4

i	C_i	i	C_i	i	C_i
-11	`0xF0`	-7	`0xB4`	-3	`0x78`
-10	`0xE1`	-6	`0xA5`	-2	`0x69`
-9	`0xD2`	-5	`0x96`	-1	`0x5A`
-8	`0xC3`	-4	`0x87`	0	`0x4B`

D.2 A 2-round Iterative Linear Trail

Table 13: 2-round iterative linear trail of Gaston with [12, 12] active S-boxes and weight 48 [24, 24].

```
......1...........1...........1...........1...1...1.......1.....
..1...........1...........1...........1...........1...1...1.....
................................................................
................................................................
................................................................
---------------------------------------------------------------- χ
......1...........1...........1...........1...1...1.......1.....
..1...........1...........1...........1...1...1.......1.........
................................................................
................................................................
................................................................
---------------------------------------------------------------- ρ_east
......1...........1...........1...........1...1...1.......1.....
......1...........1...........1...........1...1...1.......1.....
................................................................
................................................................
................................................................
---------------------------------------------------------------- θ
......1...........1...........1...........1...1...1.......1.....
......1...........1...........1...........1...1...1.......1.....
................................................................
................................................................
................................................................
---------------------------------------------------------------- ρ_west
......1...........1...........1...........1...1...1.......1.....
..1...........1...........1...........1...........1...1...1.....
................................................................
................................................................
................................................................
---------------------------------------------------------------- χ
......1...........1...........1...........1...1...1.......1.....
..1...........1...........1...........1...1...1.......1.........
................................................................
................................................................
................................................................
```

D.3 A 4-round Linear Trail

Table 14: 4-round linear trail of Gaston with [11,9,10,11] active S-boxes and weight 82 [22,18,20,22].

```
............1...1............1...1..............................
1............................1........1.......1........1.......1
................................................................
...............................1................................
................1................1............1...1............1
---------------------------------------------------------------- χ
............1................1................1.................
1.....................................1................1........
................................................................
...............................1................................
................1................1............1...1............1
---------------------------------------------------------------- ρ_east
............1................1................1.................
....1.....................................1................1....
................................................................
....1...........................................................
............1................1............1...1............1....
---------------------------------------------------------------- θ
............1................1................1.................
....1.....................................1................1....
................................................................
....1...........................................................
............1................1............1...1............1....
---------------------------------------------------------------- ρ_west
............1................1................1.................
...1........1.....................................1.............
................................................................
......................1.........................................
...1............1................1................1............1
---------------------------------------------------------------- χ
............1................1................1................1
................................................................
................................................................
......................1.........................................
...1............1................1................1............1
---------------------------------------------------------------- ρ_east
............1................1................1................1
................................................................
................................................................
...........................................................1....
............1................1................1............1...1
---------------------------------------------------------------- θ
............1................1................1................1
................................................................
................................................................
...........................................................1....
............1................1................1............1...1
---------------------------------------------------------------- ρ_west
............1................1................1................1
................................................................
................................................................
.............1..................................................
...1............1...1............1................1.............
---------------------------------------------------------------- χ
............1...1............1................1...1............1
................................................................
................................................................
.............1..................................................
...1............1...1............1................1.............
---------------------------------------------------------------- ρ_east
............1...1............1................1...1............1
................................................................
................................................................
..................................................1.............
............1...1............1................1................1
---------------------------------------------------------------- θ
............1...1............1................1...1............1
................................................................
................................................................
..................................................1.............
............1...1............1................1................1
---------------------------------------------------------------- ρ_west
............1...1............1................1...1............1
................................................................
................................................................
....1...........................................................
...1................1............1...1............1.............
---------------------------------------------------------------- χ
............1...1............1................1................1
................................................................
................................................................
....1...........................................................
...1................1............1...1............1.............
```

E Differential and Linear Trails of Gaston-S

Table 15: 3-round linear trail of Gaston-S with [51, 6, 6] active S-boxes and weight 141 [117, 12, 12].

```
1..111..1...1..1.......11...1..11.1.1....11....11.1.1......1.1..
11.1.1.1........1........1...1.11.1.1111.1.1...1.1...11.1.11.1..
1.....1...1.11.11.1..1....1.......1...1.....1.......11..1.11....
.1....1.....11.....11...1.......111111...1111.......1..1.......1
...1.11.........1....1.11.1....111111.11..1....1..11.1..1.111...
---------------------------------------------------------------- χ
1...1.1.1......1........1...1...1..1.....11.....1..11......1.1..
.1..1..11.......1........1...1.1.1..111..1.1.....1..111.1.1.....
......1...1.11....1..1...11.......1...1..11.1...1.....1.1.1.....
............1...1..111..1.1.....1..11...........1...11.1.......1
...1.11................1..1....1.11....1.1.....1..11.......11..1
---------------------------------------------------------------- ρ_east
1...1.1.1......1........1...1...1..1.....11.....1..11......1.1..
....1..1..11.......1........1...1.1.1..111..1.1.....1..111.1.1..
.....1.1.1...........1...1.11....1..1...11.......1...1..11.1...1
...1..111..1.1.....1..11...........1...11.1.......1............1
....1..1....1.11....1.1.....1..11.......11..1...1.11............
---------------------------------------------------------------- θ
................................................................
..................................1......................1......
.................................1...........1...........1......
.............................................................1..
..............1..............................................1..
---------------------------------------------------------------- ρ_west
................................................................
.....................................1......................1...
........1.......................................1...........1...
................................................1...............
..........................................1................1....
---------------------------------------------------------------- χ
................................................................
.....................................1......................1...
........1.......................................1...............
................................................................
..........................................1................1....
---------------------------------------------------------------- ρ_east
................................................................
........................................1......................1
.......................1.......................................1
................................................................
.......................1................1.......................
---------------------------------------------------------------- θ
................................................................
........................................1......................1
.......................1.......................................1
................................................................
.......................1................1.......................
---------------------------------------------------------------- ρ_west
................................................................
..1........................................1....................
..............1.......................1.........................
................................................................
....1................1..........................................
---------------------------------------------------------------- χ
................................................................
..1........................................1....................
..............1.......................1.........................
................................................................
....1................1..........................................
```

Table 16: 3-round differential trail of Gaston-S with [6, 6, 46] active S-boxes and weight 148 [12, 12, 124].

```
................................................................
...............................................1.............1..
..1................................1............................
.............1................1.................................
................................................................
---------------------------------------------------------------- χ
................................................................
...............................................1.............1..
..1................................1............................
.............1................1.................................
................................................................
---------------------------------------------------------------- ρ_east
...............................................................
1................................................1.............
................1................................1.............
1................1.............................................
...............................................................
---------------------------------------------------------------- θ
...............................................................
1.................................................1............
.................1...............................1.............
1...............1..............................................
...............................................................
---------------------------------------------------------------- ρ_west
................................................................
...1.................................................1..........
.1..............................1...............................
....1..............................................1............
................................................................
---------------------------------------------------------------- χ
................................................................
...11................................................1..........
.1..1...........................1...............................
....1..............................................1............
................................................................
---------------------------------------------------------------- ρ_east
...............................................................
.....11................................................1.......
...............1..1...........................1................
......................................1...............1........
...............................................................
---------------------------------------------------------------- θ
..........111.1.1...1...1.....1...........11.11..1...11..1.....
.......1..11.....1...1..1...1.1.......1...11........111..1.....
1..1...1...11...1..11....1....1....1.......1.1.1.....1.1..1....
.......1.....1.....1.......1.1.......11..........1...111.......
..1..1.....1.......11......1.....1.1....1.111..1....11.11....1.
---------------------------------------------------------------- ρ_west
..........111.1.1...1...1.....1...........11.11..1...11..1......
..........1..11.....1...1..1...1.1.......1...11........111..1...
.....1.1..1....1..1...1...11...1..11....1....1....1.......1.1.1.
.1.....1.......1.1.......11..........1...111..............1....1
.11......1.....1.1....1.111..1....11.11....1...1..1.....1.......
---------------------------------------------------------------- χ
...........11...1.............1...........11.....1...11.........
.............11.....1...1........1............1........111......
.....1....1.......1........1...1........1....1..............1.1.
.......1.................................1................1....1
.11......1.....1.1....1..11..1....11.11........1..1.............
```

F Differential and Linear Trails of SBD

Table 17: 3-round linear trail of SBD with [26, 6, 6] active S-boxes and weight 81.46 [53.64, 12, 15.82].

```
...........................1.11....1......1111................1.
1..........1..........1......11...1..1....1.....11.............1
...........1.......1.......1...........1.1111............11.....
..............11.............11..........1.1.....1..............
..........................11........1.....1......1..............
---------------------------------------------------------------- χ
.............................11....1.......111................1.
..........................11........1....11.1...................
...........1..11.............11..........1.......1..............
...........1.......1.......1...........1..11.............11.....
1.....................1...........1..1....1.....11.............1
---------------------------------------------------------------- ρ_east
.............................11....1.......111................1.
.............................11........1....11.1................
1.........................1..11.............11..........1.......
......1.......1...........1..11.............11................1.
...1...........1..1....1.....11.............11..................
---------------------------------------------------------------- θ
..............................................................1.
................................................................
..........................1.....................................
..........................1...................................1.
...1..............1.............................................
---------------------------------------------------------------- ρ_west
..............................................................1.
................................................................
.........................................1......................
.............1...................................1..............
................................................1..............1
---------------------------------------------------------------- χ
.................................................1.............1
................................................................
.............1..................................1...............
.........................................1....................1.
................................................................
---------------------------------------------------------------- ρ_east
.................................................1.............1
................................................................
............................1..................................1
............................1....................1..............
................................................................
---------------------------------------------------------------- θ
.................................................1.............1
................................................................
............................1..................................1
............................1....................1..............
................................................................
---------------------------------------------------------------- ρ_west
.................................................1.............1
................................................................
..............1............................1....................
...............1....................1...........................
................................................................
---------------------------------------------------------------- χ
.................................................1.............1
................................................................
...............1....................1...........................
..............1............................1....................
................................................................
```

Table 18: 3-round differential trail of SBD with [6, 6, 30] active S-boxes and weight 115 [18, 20, 82.33].

```
1.............1.....1......1....................11..............
................................................................
................................................................
................................................................
................................................................
---------------------------------------------------------------- χ
1.............1.................................................
................................................................
....................1............................1..............
...........................1....................1...............
................................................................
---------------------------------------------------------------- ρ_east
1.............1.................................................
................................................................
1..................................1............................
..............1....................1............................
................................................................
---------------------------------------------------------------- θ
1.............1.................................................
................................................................
1..................................1............................
..............1....................1............................
................................................................
---------------------------------------------------------------- ρ_west
1.............1.................................................
................................................................
...............1..................................1.............
.1....................1.........................................
................................................................
---------------------------------------------------------------- χ
11..............................................................
................................................................
......................1.........................................
..............1...................................1.............
1..............1................................................
---------------------------------------------------------------- ρ_east
11..............................................................
................................................................
.....................................1..........................
.1...................................1..........................
.............................................1..............1...
---------------------------------------------------------------- θ
11................111.....1.1....11..11...............1.........
................1.11..1.1........11..11...........1.............
.....1.1..........11..............1...1........................1
.1................1..............11...1..........1.....1.1......
..........1.......11.............11..11.1....11.1...........1...
---------------------------------------------------------------- ρ_west
11................111.....1.1....11..11...............1.........
...................1.11..1.1........11..11...........1..........
..............1.....1.1..........11..............1...1..........
.....1..............11...1..........1.....1.1.......1...........
1.............11..11.1....11.1...........1.............1.......1
---------------------------------------------------------------- χ
11...1............1..1...11.1....11.1.1...1.1....1..1.1.........
..............11...111.....1.1...11.....11.............1.......1
...................1111..1.1........1....1...........1..........
..............1....11.1..........11..1...............1..........
1.............1...11.....111........11...1......................
```

G Test Vectors

Table 19: Test vectors for 12 rounds of Gaston-S. The input and its corresponding output are on the left-side and the right-side, respectively.

0x0000000000000000	0x011A9C288266AA19
0x0000000000000000	0x8FAC076FD9C210C4
0x0000000000000000	0xCCE7C9D2584B54C9
0x0000000000000000	0xAABE797E89A042FD
0x0000000000000000	0x988E0FE8AC4A6EAA
0x806763C2DC0FBA14	0xEA74E92BBE967B58
0x0E4310C6C69C6B64	0xD0B523B86ED20BC3
0x7D3F5591E4E06046	0x91C40893C774101B
0xE40FBAC455F26EDB	0xC151A39DD4F47AEE
0xD339673CA8E6C673	0x04291BE37749886F
0xFFFFFFFFFFFFFFFF	0x1E1BBE786C2CCF54
0x0123456789ABCDEF	0x883DE3A68924F873
0xFEDCBA9876543210	0x661810FCB001170F
0xAAAAAAAAAAAAAAAA	0xA16BEE3732869D79
0x0101010101010101	0xA23027988C91D0A2

Table 20: Test vectors for 8 rounds of SBD. The input and its corresponding output are on the left-side and the right-side, respectively.

0x0000000000000000	0x03CFF13FF5463C79
0x0000000000000000	0x4545716CA205D7DF
0x0000000000000000	0x50088C8B4107FE2C
0x0000000000000000	0x6A45A68EE3CE99E0
0x0000000000000000	0xC3BEFD15D854ADC1
0x205A2B2C684B3180	0x06C4DA1150967B94
0x9A908E723D6E2186	0xF1E883EB31C34286
0x63416D5B1B618ACF	0x2F0E66616C43F638
0xD850966EDB2C1683	0x044932E45C362D68
0x20DA8EB257C417B8	0xE9C40C02A0B82C7C
0xFFFFFFFFFFFFFFFF	0x447E83F11FA7F36B
0x0123456789ABCDEF	0xCB640EDFC1889356
0xFEDCBA9876543210	0x55E3EB9E6DC7E042
0xAAAAAAAAAAAAAAAA	0x3A4CAB09A88BE5A2
0x0101010101010101	0x49FEA5E018048615