Received November 18, 2016, accepted December 19, 2016, date of publication March 28, 2017,
date of current version January 18, 2018.

Digital Object Identifier 10.1109/ACCESS.2017.2688427

Distinguishers for 4-Branch and 8-Branch Generalized Feistel Network

DONGHOON CHANG, ABHISHEK KUMAR, AND SOMITRA KUMAR SANADHYA
Computer Science, Indraprastha Institute of Information Technology, New Delhi 110020, India

Corresponding author: Abhishek Kumar (abhishekk@iiitd.ac.in)

ABSTRACT In this paper, we present an eight round distinguisher for four-branch type-2 generalized Feistel network (GFN) with double-SP (DSP) functions and two distinguishers for eight-branch type-2 GFN with single-SP (SSP) functions in a known key attack (KKA) model. We improved the result presented by Sasaki in Indocrypt 2012 by extending the number of rounds attacked from seven to eight for four-branch GFN. Furthermore, for eight-branch type-2 GFN with SSP functions, we present the first known key distinguishers. Our attack works up to 15 rounds of this GFN for all practical parameters. Subsequently, we extend the attack to 17 rounds for the same GFN, which works for most practical parameters. On the basis of our second result and the number of rounds attacked, we conclude that eight-branch type-2 GFN with SSP functions is weaker than four-branch type-two GFN with DSP functions in the KKA model. We apply rebound attack technique to mount all three distinguishers. However, a limitation of all the distinguishers presented in this paper is that they are useful only if the input size of S-boxes in bits is greater than or equal to the number of S-boxes in one S-box layer.

INDEX TERMS Rebound attack, block cipher, SP-functions, known-key distinguisher, generalized feistel network, active S-box.

I. INTRODUCTION

Design and analysis of block ciphers has been a challenging and interesting area for cryptographers. Feistel network and Generalized Fesitel Network (GFN) have been popular choices for designing block ciphers since the seminal work of Luby and Rackoff [1]. GFN are widely used for lightweight designs due to their compactness as well as other desirable implementation properties like smaller round functions in comparison to the standard Fesitel structure. Some noteworthy block ciphers based on GFN are CLEFIA [2], RC6 [3] and HIGHT [4].

Traditionally, block ciphers are used as the basic building blocks of many cryptographic primitives, e.g., signencryption schemes, compression functions and authenticated encryption schemes. The security of a block cipher depends on the round function as well as the secret key. Recently, researchers proposed many attacks in the context of known-key setting [5]–[12], where the secret key is already known to the attacker. In this case only the randomness of the key and the round function provides the security.

Generally, the round function of block ciphers use a nonlinear mapping termed as substitution transformation (S-box transformation) to create confusion followed by a linear layer termed as permutation transformation (P-box transformation) and subkey XOR-ing. Popularly termed as the SP-layer. An S-box S is called differentially 'active', if the input difference to S is non zero. Since, only the active S-boxes created confusion, in general more active S-boxes provide a more secure design against traditional differential [13] and linear attacks [14].

GFN with more branches is a matter of great interest for the design of lightweight cryptographic algorithms which are more suitable for ubiquitous computing system. Security analysis of GFN has been an interesting area for cryptographers. In 2011, Bogdanov and Shibutani [15] analyzed the security of GFNs in terms of repetition of SP-layer for two different 4-branch GFN, type-1 and type-2. They compared their findings with other known results [16], [17] for same GFN with SSP function and proved that DSP function has more active S-boxes than SSP function for the same number of S-boxes used in the designs. This is shown in Table 1. On the basis of this fact they concluded that DSP function is more secure than SSP functions against linear and differential cryptanalysis. In 2011, Sasaki and Yasuda [18]applied rebound attack [19] on 2-branch SSP functions GFN and successfully mounted known key distinguisher up to 11 rounds.

TABLE 1. Results from [15]. Comparison of number of active S-boxes for DSP function GFN and SSP function GFN.

GFN Type	r	$A_{m,r}$	$S_{m,r}$	E_m	E
type-1, SSP [16]	$16R$	$[3(m+1)+1]R$	$16mR$	$\frac{3m+4}{16m}$	$\frac{3}{16}$
type-1, DSP [15]	$14R$	$[7(m+1)+1]R$	$28mR$	$\frac{m+1}{4m}$	$\frac{1}{4}$
type-2 , SSP [17]	$6R$	$[2(m+1)+2]R$	$12mR$	$\frac{2m+3}{12m}$	$\frac{1}{6}$
type-2 , DSP [15]	$6R$	$[6(m+1)]R$	$24mR$	$\frac{m+1}{4m}$	$\frac{1}{4}$

In Indocrypt 2012, Sasaki [20] presented a 7 round known key distinguisher for 4-branch type-2 GFN with DSP functions. On the basis of number of rounds attacked, he concluded that the DSP is weaker than SSP for finite number of rounds.

Our Contribution: In this work, we present a distinguisher for 4-branch type-2 GFN with DSP functions and two distinguishers for 8-branch type-2 GFN with SSP functions. One round of 4-branch type-2 GFN with DSP functions contains 4 SP-layers. In our first work, we extend the number of rounds attacked from 7 to 8, i.e., 32 SP-layers for 4-branch type-2 GFN with DSP functions. This analysis is the first analysis for 8 rounds of the above mentioned GFN. Our result strengthens the belief of [20] that DSP function is indeed weaker than SSP function.

Further, we present two different distinguishers for 8-branch type-2 GFN with SSP functions. The number of rounds attacked using the first distinguisher is 15 and the number of rounds attacked by the second distinguisher is 17. On the basis of number of rounds attacked and the number of S-boxes used, proposed distinguiers can be used to compare security of different GFNs. In general, r rounds of a SSP GFN can be compared to $r/2$ rounds of a DSP function. Therefore, on the basis of [18] and our results, we conclude that 8-branch type-2 GFN is weaker than 4-branch type-2 GFN with DSP functions as well as 2-branch GFN with SSP function.

The rest of the paper is organized as follows. In § II, the notation used and related prior work is described. Our attack on 4-branch type-2 GFN with DSP functions is presented in § III. In § IV, we present an attack on 8-branch type-2 GFN with SSP functions which extends up to 15 rounds. This attack is valid for all practical parameters of this design. We then extend the attack further up to 17 rounds in § V, which is valid for most practical parameters of the design. Finally, we conclude the work in § VI with some open problems.

II. PRELIMINARIES

A. THE GENERALIZED FEISTEL NETWORK (GFN)

Generalized Feistel Networks (GFN) are variants of Feistel networks with more than two branches, i.e., a m-branch GFN partitions the N-bit blocks into m sub-blocks. The size of each sub-blocks is n bits, i.e., $N = n \times m$.

As defined above, an m-branch GFN divides an N-bit blocks equally in m sub-blocks such that $N = (n_1, n_2, n_3, \cdots, n_m)$ and a round of type-2 GFN outputs $[n_m \oplus f_{\frac{m}{2}}(n_{m-1}), n_1, n_2 \oplus f_1(n_1), n_3, \cdots, n_{m-2} \oplus f_{\frac{m}{2}-1}(n_{m-3}), n_{m-1}]$ for keyed nonlinear functions $f_1, f_2, \ldots, f_{\frac{m}{2}}$ [22]. Schematic diagrams of 4-line type-2 GFN and 8-line type-2 GFN are shown in Figure 1 and Figure 2 respectively.

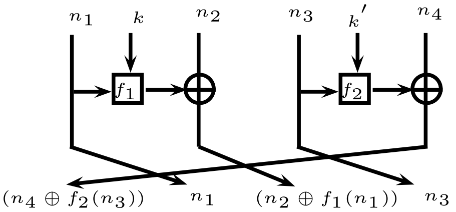

FIGURE 1. 4-branch type-2 GFN.

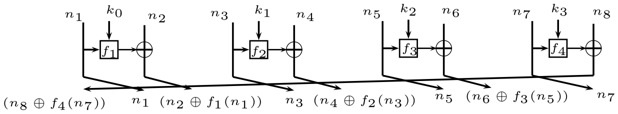

FIGURE 2. 8-branch type-2 GFN.

B. NOTATIONS

1) N: Block length in bits.
2) n: Input/output size of the sub-blocks and round functions.
3) c: Size of an S-box in bits.
4) r: Number of S-boxes in an S-box layer.
5) $(bin_n(0)$: Representation of value zero using n-bit.
6) ?: Any unknown n-bit value out of the possible 2^n.
7) $bin_c^j(a)$: Represents a value a (using c bit) which is the j-th byte of the word.
8) 1_a: Represents is the n-bit constant where only one predetermined (j-th) byte is active.

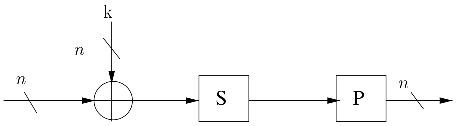

FIGURE 3. SSP round function. The numbers on the crossed branches represent the bit length of that branch.

C. SP ROUND FUNCTION

The round function of a GFN consists of these elementary operations:

1) Subkey XOR
2) Substitution Layer (S-box)
3) Permutation Layer (P-layer)

1) **Single-SP Round Function**: A SSP round function of a GFN consist these operations: subkey xoring, S-box layer followed by a permutation layer. The design of single-SP round function is described in Figure 3.
2) **Double-SP Round Function**: DSP round function is applying the single-SP round function twice one after another [15]. Figure 4 describes a double SP-round function.

III. 8-ROUND DISTINGUISHING ATTACK ON TYPE-2 4-BRANCH GFN WITH DSP FUNCTIONS

Here, we present a known-key distinguishing attack on a block cipher $E_K(\cdot)$ which is instantiated with 8-round type-2 4-branch DSP GFN.

TABLE 2. Comparison between general attack and presented distinguishers.

GFN Type	Rounds Attacked	N	n	c	r	Attack Complexity		
						general attack	same S-box	different S-box
4-branch type-2 DSP	8	64	16	4	4	$32 \times r \times 2^8 = 2^{15}$	2^8	2^{10}
		128	32	8	4	$32 \times r \times 2^{16} = 2^{23}$	2^{16}	2^{18}
		256	64	8	8	$32 \times r \times 2^{32} = 2^{40}$	2^{16}	2^{19}
8-branch type-2 SSP	15	64	8	4	2	$60 \times r \times 2^6 \approx 2^{13}$	2^8	2^9
		128	16	8	2	$60 \times r \times 2^{12} \approx 2^{20}$	2^{16}	2^{17}
			16	4	4	$60 \times r \times 2^{14} \approx 2^{22}$	2^8	2^{10}
		256	32	8	4	$60 \times r \times 2^{28} \approx 2^{34}$	2^{16}	2^{18}
	17	64	8	4	2	$68 \times r \times 2^2 \approx 2^9$	2^8	2^9
		128	16	4	4	$68 \times r \times 2^6 \approx 2^{14}$	2^8	2^{10}
		256	32	8	4	$68 \times r \times 2^{12} \approx 2^{21}$	2^{16}	2^{18}

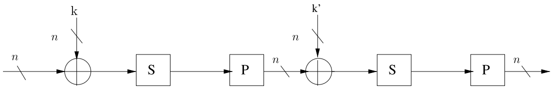

FIGURE 4. DSP round function. The numbers on the crossed branches represent the bit length of that branch.

Theorem 1 (Our Result): Let $E_K(\cdot)$ be a block cipher, which is 8-round type-2 4-branch DSP GFN, with block size $N(= 4n)$, where K is a randomly chosen and public N-bit key. For any given c-bit constant a, we show that we can find a message pair (M, M') and the corresponding ciphertext pair (C, C') with complexity 2^{2c}(same S-box) or $r \times 2^{2c}$ (different S-box) such that

- $M \oplus M' = (bin_n(0), X, ?, ?)$, and
- $C \oplus C' = (?, ?, P[1_a], ?)$,

where $C = E_K(M)$, $C' = E_K(M')$, P is the underlying $r \times r$ MDS matrix of the block cipher E,[1] and X is any full-active difference.[2]

For general attack, over N bits, we show that it requires an effort of $32 \times r \times 2^{n/2}$ to find such plaintext and ciphertext pairs where the effort is measured in terms of the number of S-box operations.

Proof 1:

a: In the Case of General Attack

For general attack, given any two messages (M, M') such that $M \oplus M' = (bin_n(0), X, ?, ?)$, the probability that the corresponding ciphertext (C, C') satisfies $C \oplus C' = (?, ?, P[1_a], ?)$ is 2^{-n} where n is the size of a word in bits. Therefore, we need $2^{n/2}$ message-ciphertext pairs to get one such message pair.

Since each round is using four S-box layers and each layer contains r S-boxes, the total number of S-boxes used are $32 \times r$. Therefore the effort of calculating $2^{n/2}$ message-ciphertext pairs can be described as $32 \times r \times 2^{n/2}$ S-box operations.

b: In the Case of $E_K(\cdot)$

The aim of our attack is to produce a pair of messages having differences of the form $(0, X, ?, ?)$ such that they produce an output difference of the form $(?, ?, P[1_a], ?)$ for all known keys.

The truncated differential characteristic followed by our 8-round attack is as follows.

$$\begin{aligned}(0, X, ?, ?) &\xrightarrow{1^{st}R} (0, 0, X, ?) \xrightarrow{2^{nd}R} (0, 0, 0, X) \xrightarrow{3^{rd}R}\\ (X, 0, 0, 0) &\xrightarrow{4^{th}R} (0, X, P[1_\alpha], 0) \xrightarrow{5^{th}R} (P[1_a], 0, X, P[1_\alpha])\\ &\xrightarrow{6^{th}R} (0, P[1_a], ?, X) \xrightarrow{7^{th}R} (?, 0, P[1_a], ?)\\ &\xrightarrow{8^{th}R} (?, ?, P[1_a], ?).\end{aligned}$$

The complexity of the proposed attack is only in finding a pair of values satisfying truncated differential path of the three rounds inbound phase. Backward outbound phase of first three rounds and forward outbound phase of the last two rounds propagate with probability one after getting any suitable pair by inbound phase.

Our differential trail starts from three round inbound phase from the 4^{th} round to the 6^{th} round, and message pairs with zero differences in the last three word and we get a difference in the first word in the middle of inbound phase. The complexity of getting a starting point (a pair of value following the inbound phase) is 2^{2c} (same S-box) or $r \times 2^{2c}$ (different S-box) both time and memory.

Given such a starting point, we get a plaintext-ciphertext pair after applying three round backward outbound phase (from 1^{st} round to the 3^{rd} round) and two round forward outbound phase (7^{th} round to the 8^{th} round). The differences in the first two words of the plaintexts are 0 and X, while the differences in the next two words could be any value. Corresponding to this pair of plaintexts, the third word of the ciphertexts will have a specific difference out of the possible $2^n - 1$ non-zero values. Note that the differences in the other three word of the ciphertext can be any possible n-bit values.

[1] the branch number of P is $(r + 1)$

[2] j is a predetermined byte, i.e., any specific byte out of r bytes.

That is, we show that we can get any 2^c specific differences out of the possible 2^n differences. Our work shows that we can create such a differential trail with lower complexity in comparison to a general attack. This can be used as a valid distinguisher against the cipher.

c: Three-Round Inbound Phase

The truncated differential trail for inbound phase propagates as:

$$(X, 0, 0, 0) \xrightarrow{4^{th}R} (0, X, P[1_\alpha], 0) \xrightarrow{5^{th}R} (P[1_a], 0, X, P[1_\alpha]) \xrightarrow{6^{th}R} (0, P[1_a], ?, X).$$

The complexity of inbound phase is the cost of finding a pair of values which follows the truncated differential path as shown in Figure 5.

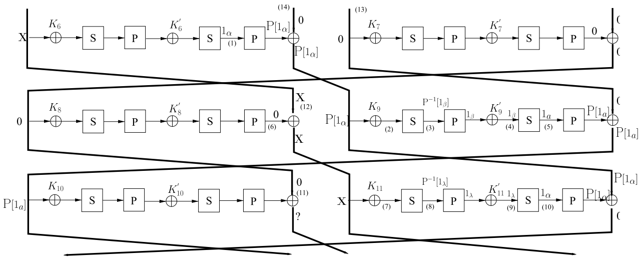

FIGURE 5. The inbound phase of our 8-round distinguisher on 4-branch type-2 GFN with double-SP functions.

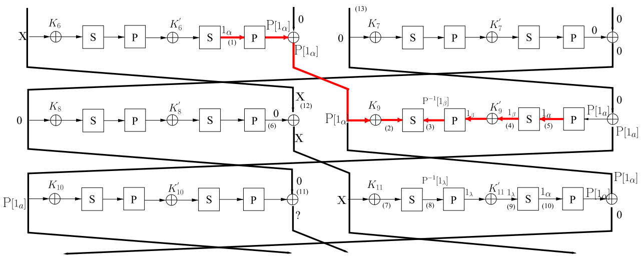

FIGURE 6. The inbound phase of 8-round distinguisher. Red colored path represents first matching of the Inbound phase.

To get inbound phase with desired differential characteristics these steps are followed.

1) Make difference distribution table (DDT) for all S-boxes.
2) a) See Figure 6. Fix the difference at position (5) as 1_a, where a is an already provided value at the beginning of the attack process as mentioned in our result (Theorem 1). Using DDT, choose any matched β such that differences 1_β and 1_a at positions (4) and (5) are matched.
   b) Since P is linear, the difference at position (3) will be $P^{-1}[1_\beta]$.
   c) For every α such that $1 \leq \alpha \leq 2^c - 1$, we define the difference at position (1) as 1_α, and repeat the following procedure.
      i) Since the difference at position (1) is 1_α, the difference at the position (2) will be $P[1_\alpha]$.

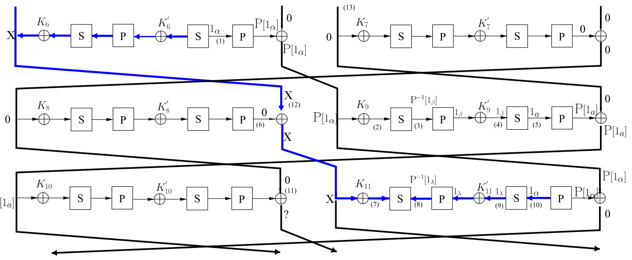

FIGURE 7. The inbound phase of 8-round distinguisher. Blue colored path represents second matching of the Inbound phase.

      ii) If the differences at positions (2) and (3) are matched, for each matched difference we can generate 2^r possible matching massage pairs at position (2) from the DDT, where the difference of each matching pair (M, M') is $P[1_\alpha]$ and the difference of $(S[M], S[M'])$ is $P^{-1}[1_\beta]$.
      iii) For each matching pair (M, M') of 2^r pairs, repeat the following procedure.
         A) If the difference of $S[P[S[M]] \oplus K'_9]$ and $S[P[S[M']] \oplus K'_9]$ at position (5) is 1_a, then fix it and fix any value for second word at position (14) with the knowledge of difference. All the corresponding values in between position (1) and (12) can be fixed from the knowledge of (M, M'). Find difference X at position (12) from the knowledge of (M, M'), and go to Step 2-(d).
         B) Else if there remain matched pairs we have not considered, go to Step 2-(c)-iii, otherwise go to Step 2-(c).
   d) Figure 7. Fix the difference at position (10) as 1_α, where α is already fixed in Step 2-(c).
   e) For every λ such that $1 \leq \lambda \leq 2^c - 1$, repeat the following procedure.
      i) Calculate the difference $P^{-1}[1_\lambda]$ at position (8). Note that we already know that the difference at position (7) should be X.
      ii) If the differences at positions (7) and (8) are matched, we generate 2^r possible matching pairs at position (7) from the DDT, where the difference of each matching pair (W, W') is X and the difference of $(S[W], S[W'])$ is $P^{-1}[1_\lambda]$.
      iii) For each matching pair (W, W') of 2^r pairs, repeat the following procedure.
         A) If the difference of S[P[S[W]]$\oplus K'_{11}$] and $S[P[S[W']] \oplus K'_9]$ at position (10) is 1_α, then fix it and calculate the value at position (6) from the knowledge of (M, M') and (W, W'), fix the value at position (13) and stop the inbound phase and exit.

B) Else if there remain matched pairs we have not considered, go to Step 2-(e)-iii, otherwise go to Step 2-(e).

d: Complexity Calculations for the Inbound Phase

Next, we provide the time and memory complexity for the procedure described above in a step by step fashion.

1) a). If the S-box layer has same S-boxes, Step 1 requires 2^{2c} time and 2^{2c} memory.
   b). If the S-box layer has all different S-boxes, Step 1 requires $r \times 2^{2c}$ time as well as memory.
2) Step 2-(a) requires constant complexity by DDT look up.
3) Step 2-(b) requires only one operation, so the complexity is constant.
4) Step 2-(c)-i again requires constant complexity.
5) Step 2-(c)-ii Since we are using r S-boxes in the S-box layer, the matching probability is 2^{-r}. And after getting one matched difference we have 2^r matching pair, so the complexity of Step 2-(c)-ii is 2^r.
6) Step 2-(c)-iii requires 2^c complexity, since after completion of this step we found all 2^{c-r} matched differences and used all 2^c matching pairs.
7) Step 2-(d) requires constant complexity.
8) Step 2-(e)-i again requires constant complexity.
9) Step 2-(e)-ii Since we are using r S-boxes in the S-box layer the matching probability is 2^{-r}. After getting one matched difference, we have 2^r matching pairs, so the complexity of Step 2-(e)-ii is 2^r.
10) Step 2-(e)-iii requires 2^c complexity. After completion of this step we found all 2^{c-r} matched differences and used all 2^c matching pairs.
11) Overall complicity of finding one starting point, i.e., one message pair, which may follow entire differential characteristic of inbound phase is 2^{2c} S-box operations (same S-box) $r \times 2^{2c}$ S-box operations (different S-box), both in terms of time and memory.

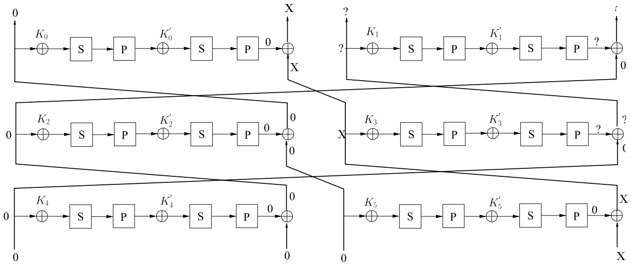

FIGURE 8. **3-round backward outbound phase.**

e: Three-Round Backward Outbound Phase

The truncated differential path of backward outbound phase is as follows, (Figure 8).

$$(0, X, ?, ?) \xrightarrow{1^{st}} (0, 0, X, ?) \xrightarrow{2^{nd}} (0, 0, 0, X) \xrightarrow{3^{rd}} (X, 0, 0, 0).$$

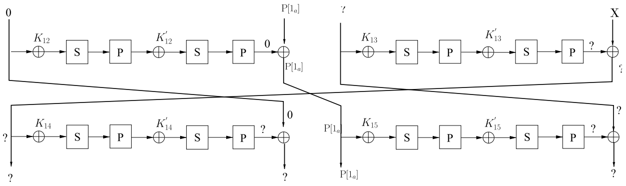

FIGURE 9. **2-round forward outbound phase.**

After getting one starting point for inbound phase, the truncated differential path for backward outbound phase propagates with probability 1.

f: Two-Round Forward Outbound Phase

The truncated differential path of forward outbound phase is as follows (Figure 9).

$$(0, P[1_a], ?, X) \xrightarrow{7^{th}} (?, 0, P[1_a], ?) \xrightarrow{8^{th}} (?, ?, P[1_a], ?).$$

Similar to the backward outbound phase, after getting any paired values which satisfies the inbound phase, the truncated differential path follows the forward outbound phase with probability 1.

TABLE 3. **Comparison of complexities between general attack and proposed distinguisher.**

N	n	c	r	complexity for general attack	complexity of our attack	
					same S-box	different S-box
64	16	4	4	2^{15}	2^8	2^{10}
128	32	8	4	2^{23}	2^{16}	2^{18}
256	64	8	8	2^{40}	2^{16}	2^{19}

A. SUMMARY OF THE ATTACK

As shown in Table 3, the complexity of the proposed distinguisher is much lower than a general attack. The proposed distinguisher is valid only if the input size of S-boxes is greater than or equal to the number of S-boxes used.

IV. 15-ROUND DISTINGUISHING ATTACK ON 8-BRANCH GFN WITH SINGLE-SP FUNCTIONS

Here, we present a new known-key distinguishing attack on a block cipher $E_K(\cdot)$ which is instantiated with 15-round 8-branch SSP GFN. Note that this is the first known attack on this GFN.

Theorem 2 (Our Result): Let $E_K(\cdot)$ be a block cipher, which is 15-round 8-branch single-SP GFN with block size $N(= 8n)$, where K is a randomly chosen and public N-bit key. We show that we can find a message pair (M, M') and the corresponding ciphertext pair (C, C') and c-bit constants a, b and d, with complexity 2^{2c} (same S-box) or $r \times 2^{2c}$ (different S-box) such that

- $M \oplus M' = (1_a, P[1_b], ?, ?, ?, ?, ?, ?)$, and
- $C \oplus C' = (?, ?, ?, ?, ?, ?, 1_a, P[1_d])$,

where $C = E_K(M)$, $C' = E_K(M')$, P is the underlying $r \times r$ MDS matrix of the block cipher E.[3]

[3] j is a predetermined byte.

On the other hand, in case of a general attack over N bits, we show that it requires a complexity of $60 \times r \times 2^{(n-\frac{c}{2})}$ to find such plaintext and ciphertext pairs where the complexity is measured in terms of the number of S-box operations. Therefore, our construction works as a distinguisher for 15 round 8-branch GFN.

Proof 2:

g: In the Case of General Attack

For a general attack, given any two messages (M, M') such that $M \oplus M' = (1_a, P[1_b], ?, ?, ?, ?, ?, ?)$, for all a and b, the probability that there exists a d such that the corresponding ciphertext (C, C') satisfies $C \oplus C' = (?, ?, ?, ?, ?, ?, 1_a, P[1_d])$ is $2^{(2n-c)}$, where n is the size of a word in bits. Therefore, we need $2^{(n-\frac{c}{2})}$ plaintext-ciphertext pairs to get one conforming message pair.

Since, each round of 8-branch type-2 GFN contains four S-box layers and each layer contains r S-boxes, the total number of S-boxes used are $60 \times r$. Therefore, the complexity of processing $2^{(n-\frac{c}{2})}$ plaintext-ciphertext pairs can be described as the time complexity of $60 \times r \times 2^{(n-\frac{c}{2})}$ S-box operations.

h: In Case of $E_K(\cdot)$

The aim of the this attack is to produce a pair of messages having differences of the form $(P[1_a], P[1_b], ?, ?, ?, ?, ?, ?)$, such that they produce an output difference of the form $(?, ?, ?, ?, ?, ?, 1_a, P[1_d])$ for all known keys.

The truncated differential characteristic followed by our 15-round attack is as follows.

$$
\begin{aligned}
&(1_a, P[1_b], ?, ?, ?, ?, ?, ?)\\
&\xrightarrow{1^{st}R}\ (0, 1_a, P[1_\delta], ?, ?, ?, ?)\ \xrightarrow{2^{nd}R}\\
&(0, 0, 1_a, P[1_\delta], ?, ?, ?, ?)\\
&\xrightarrow{3^{rd}R}\ (0, 0, 0, 1_a, P[1_\eta], ?, ?, ?)\\
&\xrightarrow{4^{th}R}\ (0, 0, 0, 0, 1_a, P[1_\eta], ?, ?)\\
&\xrightarrow{5^{th}R}\ (0, 0, 0, 0, 0, 1_a, P[1_\alpha], ?)\\
&\xrightarrow{6^{th}R}\ (0, 0, 0, 0, 0, 0, 1_a, P[1_\alpha]) \xrightarrow{7^{th}R} (0, 0, 0, 0, 0, 0, 0, 1_a)\\
&\xrightarrow{8^{th}R}\ (1_a, 0, 0, 0, 0, 0, 0, 0) \xrightarrow{9^{th}R} (0, 1_a, P[1_\beta], 0, 0, 0, 0, 0)\\
&\xrightarrow{10^{th}R}\ (0, 0, 1_a, P[1_\beta], 1_\gamma, 0, 0, 0)\ \xrightarrow{11^{th}R}\\
&(0, 0, 0, 1_a, 0, 1_\gamma, P[1_\phi], 0)\\
&\xrightarrow{12^{th}R}\ (?, 0, 0, 0, 1_a, 0, 1_\gamma, P[1_\phi])\\
&\xrightarrow{13^{th}R}\ (P[1_\theta], ?, ?, ?, 0, 1_a, P[1_\lambda], 1_\gamma)\ \xrightarrow{14^{th}R}\\
&(?, P[1_\theta], ?, ?, ?, 0, 1_a, P[1_\lambda])\\
&\xrightarrow{15^{th}R}\ (?, ?, ?, ?, ?, ?, 1_a, P[1_d]).
\end{aligned}
$$

The complexity of the proposed attack is only in finding a pair of values satisfying truncated differential path of the four rounds inbound phase. Backward outbound phase of first seven rounds and forward outbound phase of the last four rounds satisfy with probability one after getting any suitable pair by inbound phase.

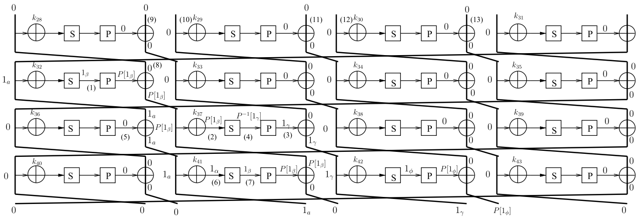

FIGURE 10. Four round inbound phase of our 15-round distinguisher on 8-branch type-2 single-SP GFN.

Our differential trail starts from four round inbound phase (8^{th} round to the 11^{th} round), with a specific difference in the last word and any arbitrary differences in the remaining seven words. The complexity of getting a starting point (a pair of values following inbound phase) is 2^{2c} (same S-box) or $r \times 2^{2c}$ (different S-box) both time and memory.

Given such a starting point, we get a plaintext-ciphertext pair after applying seven round backward outbound phase (1^{st} round to the 7^{th} round) and four round forward outbound phase (12^{th} round to the 15^{th} round). The difference in the 1st and 2nd word of plaintexts and 7th and 8th word of the ciphertexts can be any fixed values out of the possible 2^c values.

Our work shows that we can create such a differential trail with lower complexity in comparison to a general attack. This can be used as a valid distinguisher against the cipher.

i: Four-Round Inbound Phase

The truncated differential trail for inbound phase is propagated as:

$$
\begin{aligned}
&(0, 0, 0, 0, 0, 0, 0, 1_a) \xrightarrow{8^{th}R} (1_a, 0, 0, 0, 0, 0, 0, 0) \xrightarrow{9^{th}R}\\
&(0, 1_a, P[1_\beta], 0, 0, 0, 0, 0) \xrightarrow{10^{th}R} (0, 0, 1_a, P[1_\beta], 1_\gamma, 0, 0, 0)\\
&\xrightarrow{11^{th}R} (0, 0, 0, 1_a, 0, 1_\gamma, P[1_\phi], 0).
\end{aligned}
$$

The complexity of inbound phase is the cost of finding a pair of values which follows truncated differential path as shown in Figure 10.

To get inbound phase with the desired differential characteristics, these steps are followed.

1) Make difference distribution table (DDT) for all S-boxes.
2) Fix a byte position j in a word to be activated, where $1 \leq j \leq r$.
3) See Figure 10. Fix the difference at position (3) as 1_γ, where γ is any nonzero c-bit value.
4) Compute the corresponding difference at position (4). Since P is linear, the difference at position (4) will be $P^{-1}[1_\gamma]$.
5) For every β such that $1 \leq \beta \leq 2^c - 1$, fix the difference at position (1) as 1_β, and repeat the following procedure.
    a) Since P is linear and difference at position (8) is 0, the difference at position (2) comes out to be $P[1_\beta]$.

b) i) If the differences at positions (2) and (4) are matched, for each matched difference we can generate 2^r possible matching massage pairs at position (2) from the DDT, where the difference of each matching pair (M, M') is $P[1_\beta]$ and the difference of (S$[M]$,S$[M']$) is $P^{-1}[1_\gamma]$, and go to Step 2-(c).

ii) Else if there are remaining β at position (1), go to Step 5.

c) Fix the constant values at position (5) and for each matching pair $(M,\ M')$ of 2^r pairs, repeat the following procedure.

i) Fix the constant value for entire word except j-byte at position (8), for every 2^c different values for j-th byte, repeat the following procedure.

A) Calculate the corresponding values (W, W') at the position (6) and if the difference at position (7) is 1_β, i.e., $(S[W] \oplus S[W']) = 1_\beta$, then fix it. Using the knowledge of (M, M') and (W, W'), fix all the corresponding values at position (1), (2), (3), (4), (6) and (7), and go to Step 6.

B) Else if there are remaining values we have not considered, go to Step 2-(c)-i.

6) Fix the constant values at positions (9), (10), (11), (12) and (13) with the knowledge of known differences.

j: Complexity Calculation for the Inbound Phase

Next, we provide the time and memory complexity for the procedure described above in a step by step fashion.

1) a). If the S-box layer has identical S-boxes, Step 1 requires 2^{2c} time and 2^{2c} memory.
b). If the S-box layer has all different S-boxes, Step 1 requires $r \times 2^{2c}$ time as well as memory.
2) Step 2 requires only one operation, so the complexity is constant.
3) Step 3 requires only one operation, so the complexity is constant.
4) Step 4 again requires constant complexity.
5) Step 5-(a) again requires constant complexity.
6) Step 2-(b) Since we are using r S-boxes in the S-box layer, the matching probability is 2^{-r}. After getting one matched difference we have 2^r matching pair, so the complexity of Step 2-(b) is 2^r.
7) Step 2-(c) Since we are looking for a fixed difference at position (7) and we have 2^c different possible differences, the complexity of Step 2-(c) is 2^c.
8) Step 2-(6) requires constant complexity.
9) Overall complexity of finding one starting point, i.e., one message pair, which follows the complete differential characteristics of inbound phase is 2^{2c} S-box operations (same S-box) or $r \times 2^{2c}$ S-box operations (different S-box), both in terms of time and memory.

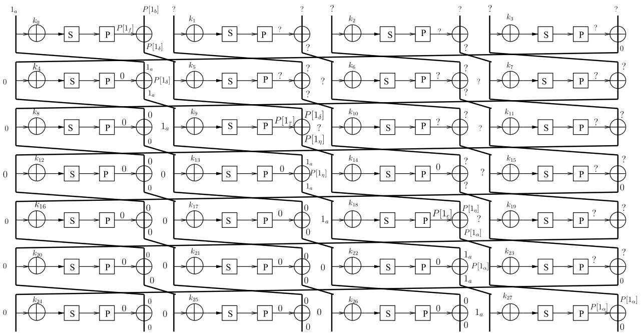

FIGURE 11. Seven round backward outbound phase of 15-round distinguisher.

k: Seven-Round Backward Outbound Phase

The truncated differential path of backward outbound phase is as follows (Figure 11).

$$(1_a, P[1_b], ?, ?, ?, ?, ?, ?) \xrightarrow{1^{st}R} (0, 1_a, P[1_\delta], ?, ?, ?, ?, ?) \xrightarrow{2^{nd}R}$$
$$(0, 0, 1_a, P[1_\delta], ?, ?, ?, ?) \xrightarrow{3^{rd}R} (0, 0, 0, 1_a, P[1_\eta], ?, ?, ?)$$
$$\xrightarrow{4^{th}R} (0, 0, 0, 0, 1_a, P[1_\eta], ?, ?)$$
$$\xrightarrow{5^{th}R} (0, 0, 0, 0, 0, 1_a, P[1_\alpha], ?)$$
$$\xrightarrow{6^{th}R} (0, 0, 0, 0, 0, 0, 1_a, P[1_\alpha])$$
$$\xrightarrow{7^{th}R} (0, 0, 0, 0, 0, 0, 0, 1_a).$$

After getting one starting point for inbound phase, the truncated differential path for backward outbound phase propagates with probability 1.

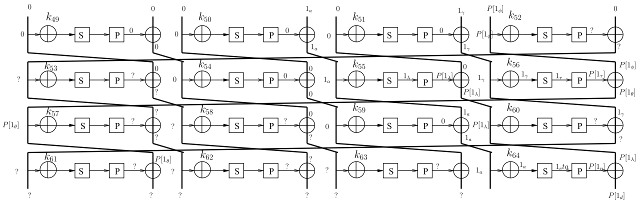

FIGURE 12. Four round forward outbound phase of 15-round distinguisher.

l: Four-Round Forward Outbound Phase

The truncated differential path of forward outbound phase is as follows, (Figure 12).

$$(0, 0, 0, 1_a, 0, 1_\gamma, P[1_\phi], 0)$$
$$\xrightarrow{12^{th}R} (?, 0, 0, 0, 1_a, 0, 1_\gamma, P[1_\phi])$$
$$\xrightarrow{13^{th}R} (P[1_\theta], ?, ?, ?, 0, 1_a, P[1_\lambda], 1_\gamma) \xrightarrow{14^{th}R}$$
$$(?, P[1_\theta], ?, ?, ?, 0, 1_a, P[1_\lambda])$$
$$\xrightarrow{15^{th}R} (?, ?, ?, ?, ?, ?, 1_a, P[1_d]).$$

Similar to the backward outbound phase, after getting any paired values which satisfies the inbound phase, the truncated differential path follows the forward outbound phase with probability 1.

TABLE 4. Comparison of complexities of general attack and our distinguisher.

N	n	c	r	complexity for general attack	complexity of our attack	
					same S-box	different S-box
64	8	4	2	$60 \times r \times 2^6 \approx 2^{13}$	2^8	2^9
128	16	8	2	$60 \times r \times 2^{12} \approx 2^{20}$	2^{16}	2^{17}
	16	4	4	$60 \times r \times 2^{14} \approx 2^{22}$	2^8	2^{10}
256	32	8	4	$60 \times r \times 2^{28} \approx 2^{34}$	2^{16}	2^{18}

A. SUMMARY OF THE ATTACK

As shown in Table 4, the complexity of the proposed distinguisher is much lower than a general attack.

Similar to the previous case, we have presented the complexity of our attack in terms of the number of S-box operations. The complexity of our proposed distinguisher is 2^{2c} (same S-box) $r \times 2^{2c}$ (different S-box) in time and memory. For general attack, the required complexity is $60 \times r \times 2^{(n-\frac{c}{2})}$ S-box operations. Therefore the proposed distinguisher has lower complexity than general attack.

V. 17-ROUND DISTINGUISHING ATTACK ON 8-BRANCH GFN WITH SINGLE-SP FUNCTIONS

In this section, we present a new known-key distinguishing attack on a block cipher $E_K(\cdot)$ which is instantiated with 17-round 8-branch single-SP GFN.

Theorem 3 (Our Result): Let $E_K(\cdot)$ be a block cipher, which is 17-round 8-branch single-SP GFN with block size $N(= 8n)$, where K is a randomly chosen and public N-bit key. We show that we can find a message pair (M, M') corresponding ciphertext pair (C, C') and c-bit constants b and d, with complexity 2^{2c} (same S-box) or $r \times 2^{2c}$ (different S-box) such that

- $M \oplus M' = (P[1_b], ?, ?, ?, ?, ?, ?, ?)$, and
- $C \oplus C' = (P[1_d], ?, ?, ?, ?, ?, ?, ?)$,

where, $C = E_K(M)$, $C' = E_K(M')$, P is the underlying $r \times r$ MDS matrix of the block cipher E.[4]

On the other hand, in case of a General Attack over N bits, we show that it requires an effort of $68 \times r \times 2^{\frac{n-c}{2}}$ S-box operations to find such plaintext and ciphertext pairs.

Proof 3:

m: In Case of General Attack

For a general attack, given any two messages (M, M') such that $M \oplus M' = (P[1_b], ?, ?, ?, ?, ?, ?, ?)$, for all y the probability that there exists a z, such that the corresponding ciphertext (C, C') satisfies $C \oplus C' = (P[1_d], ?, ?, ?, ?, ?, ?, ?)$ is $2^{(n-c)}$, where n is the size of a word in bits. Therefore, we need $2^{\frac{n-c}{2}}$ plaintext-ciphertext pairs to get one conforming message pair. The complexity of processing $2^{(n-\frac{c}{2})}$ plaintext-ciphertext pairs can be described as the time complexity of $68 \times r \times 2^{\frac{n-c}{2}}$ S-box operations.

n: In Case of $E_K(\cdot)$

The aim of the our attack is to produce a pair of messages having differences of the form $(P[1_b], ?, ?, ?, ?, ?, ?, ?)$, such that they produce an output difference of the form $(P[1_d], ?, ?, ?, ?, ?, ?, ?)$ for all known keys.

The truncated differential characteristic followed by our 17-round attack is as follows.

$$(P[1_b], ?, ?, ?, ?, ?, ?, ?)$$

$$\xrightarrow{1^{st}R} (1_a, P[1_b], ?, ?, ?, ?, ?, ?) \xrightarrow{2^{nd}R}$$

$$(0, 1_a, P[1_\delta], ?, ?, ?, ?)$$

$$\xrightarrow{3^{rd}R} (0, 0, 1_a, P[1_\delta], ?, ?, ?, ?) \xrightarrow{4^{th}R}$$

$$(0, 0, 0, 1_a, P[1_\eta], ?, ?, ?)$$

$$\xrightarrow{5^{th}R} (0, 0, 0, 0, 1_a, P[1_\eta], ?, ?)$$

$$\xrightarrow{6^{th}R} (0, 0, 0, 0, 0, 1_a, P[1_\alpha], ?)$$

$$\xrightarrow{7^{th}R} (0, 0, 0, 0, 0, 0, 1_a, P[1_\alpha])$$

$$\xrightarrow{8^{th}R} (0, 0, 0, 0, 0, 0, 0, 1_a) \xrightarrow{9^{th}R}$$

$$(1_a, 0, 0, 0, 0, 0, 0, 0)$$

$$\xrightarrow{10^{th}R} (0, 1_a, P[1_\beta], 0, 0, 0, 0, 0) \xrightarrow{11^{th}R}$$

$$(0, 0, 1_a, P[1_\beta], 1_\gamma, 0, 0, 0)$$

$$\xrightarrow{12^{th}R} (0, 0, 0, 1_a, 0, 1_\gamma, P[1_\phi], 0)$$

$$\xrightarrow{13^{th}R} (?, 0, 0, 0, 1_a, 0, 1_\gamma, P[1_\phi]) \xrightarrow{14^{th}R}$$

$$(P[1_\theta], ?, ?, ?, 0, 1_a, P[1_\lambda], 1_\gamma) \xrightarrow{15^{th}R}$$

$$(?, P[1_\theta], ?, ?, ?, 0, 1_a, P[1_\lambda]) \xrightarrow{16^{th}R}$$

$$(?, ?, ?, ?, ?, ?, 1_a, P[1_d])$$

$$\xrightarrow{17^{th}R} (P[1_d], ?, ?, ?, ?, ?, ?, ?).$$

Our differential trail starts from four round inbound phase (9^{th} round to the 12^{th} round), with a specific difference in the last word and any arbitrary differences in the remaining seven words. The complexity of getting a starting point (a pair of value following inbound phase) is $2^{[}2c]$ (same S-box) or $r \times 2^{2c}$ (different S-box) time and memory both.

Given such a starting point, we can get a plaintext-ciphertext pair after applying eight round backward outbound phase (1^{st} round to 8^{th} round) and five round forward outbound phase (13^{th} round to 17^{th} round). The difference in the first word of plaintext and ciphertext can be any fixed value out of the possible $2^c - 1$ values. That is, we show that we can get any $2^c - 1$ specific differences out of the possible 2^n differences.

Our work shows that we can create such a differential trail with lower complexity in comparison to a general attack. This can be used as a valid distinguisher against the cipher.

[4] j is a predetermined byte, i.e. any specific byte out of r bytes.

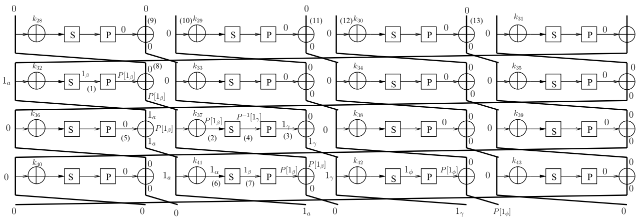

FIGURE 13. Four round inbound phase of our 17-round distinguisher on 8-branch type-2 single-SP GFN.

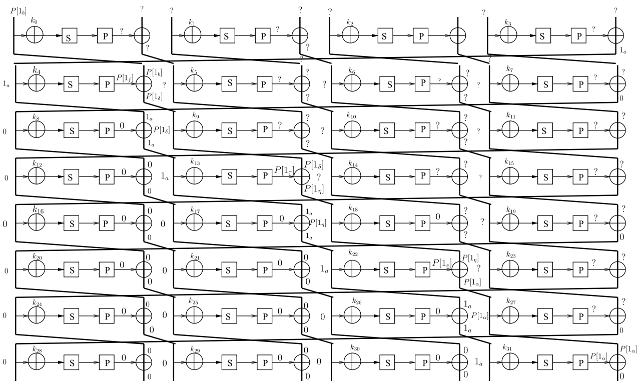

FIGURE 14. Eight round backward outbound phase of our 17-round distinguisher on 8-branch type-2 single-SP GFN.

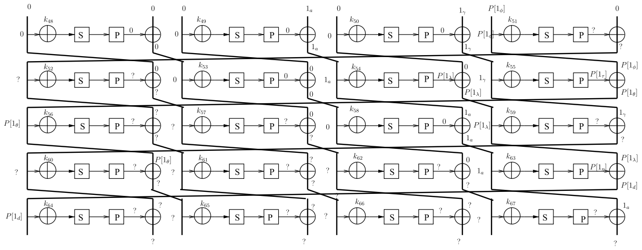

FIGURE 15. Five round forward outbound phase of our 17-round distinguisher on 8-branch type-2 single-SP GFN.

o: Four-Round Inbound Phase

The truncated differential trail for inbound phase propagates as:

$$(0, 0, 0, 0, 0, 0, 0, 1_a) \xrightarrow{9^{th}R} (1_a, 0, 0, 0, 0, 0, 0, 0) \xrightarrow{10^{th}R}$$
$$(0, 1_a, P[1_\beta], 0, 0, 0, 0, 0) \xrightarrow{11^{th}R} (0, 0, 1_a, P[1_\beta], 1_\gamma, 0, 0, 0)$$
$$\xrightarrow{12^{th}R} (0, 0,0, 1_a, 0, 1_\gamma, P[1_\phi], 0).$$

The complexity of inbound phase is the cost of finding a pair of values which follows the truncated differential path as shown in Figure 13.

Since the same four round inbound phase used in section IV is being used, the overall complexity of finding one starting point, i.e. one message pair, which follows the complete differential characteristics of the inbound phase is 2^{2c} (same S-box) or $r \times 2^{2c}$ (different S-box) both in terms of time and memory.

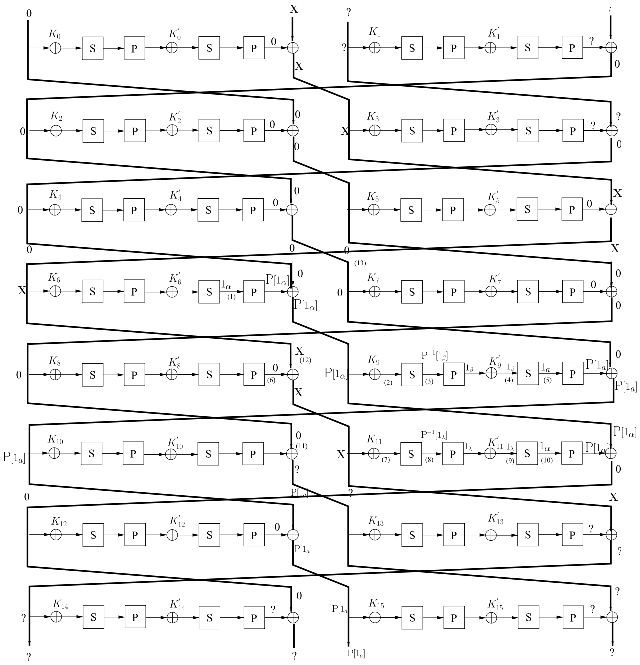

FIGURE 16. Differential path of eight round distinguisher.

p: Eight-Round Backward Outbound Phase

The truncated differential path of backward outbound phase is as follows, (Figure 14).

$$(P[1_b], ?, ?, ?, ?, ?, ?, ?)$$
$$\xrightarrow{1^{st}R} (1_a, P[1_b], ?, ?, ?, ?, ?, ?) \xrightarrow{2^{nd}R}$$
$$(0, 1_a, P[1_\delta], ?, ?, ?, ?, ?)$$
$$\xrightarrow{3^{rd}R} (0, 0, 1_a, P[1_\delta], ?, ?, ?, ?) \xrightarrow{4^{th}R}$$
$$(0, 0, 0, 1_a, P[1_\eta], ?, ?, ?)$$
$$\xrightarrow{5^{th}R} (0, 0, 0, 0, 1_a, P[1_\eta], ?, ?)$$
$$\xrightarrow{6^{th}R} (0, 0, 0, 0, 0, 1_a, P[1_\alpha], ?) \xrightarrow{7^{th}R}$$
$$(0, 0, 0, 0, 0, 0, 1_a, P[1_\alpha])$$
$$\xrightarrow{8^{th}R} (0, 0, 0, 0, 0, 0, 0, 1_a).$$

After getting one starting point for the inbound phase, the truncated differential path for the backward outbound phase propagates with probability 1.

q: Five-Round Forward Outbound Phase

The truncated differential path of forward outbound phase is as follows, (Figure 15).

$$(0, 0, 0, 1_a, 0, 1_\gamma, P[1_\phi], 0)$$
$$\xrightarrow{13^{th}R} (?, 0, 0, 0, 1_a, 0, 1_\gamma, P[1_\phi])$$
$$\xrightarrow{14^{th}R} (P[1_\theta], ?, ?, ?, 0, 1_a, P[1_\lambda], 1_\gamma) \xrightarrow{15^{th}R}$$
$$(?, P[1_\theta], ?, ?, ?, 0, 1_a, P[1_\lambda])$$
$$\xrightarrow{16^{th}R} (?, ?, ?, ?, ?, ?, 1_a, P[1_d])$$
$$\xrightarrow{17^{th}R} (P[1_d], ?, ?, ?, ?, ?, ?, ?).$$

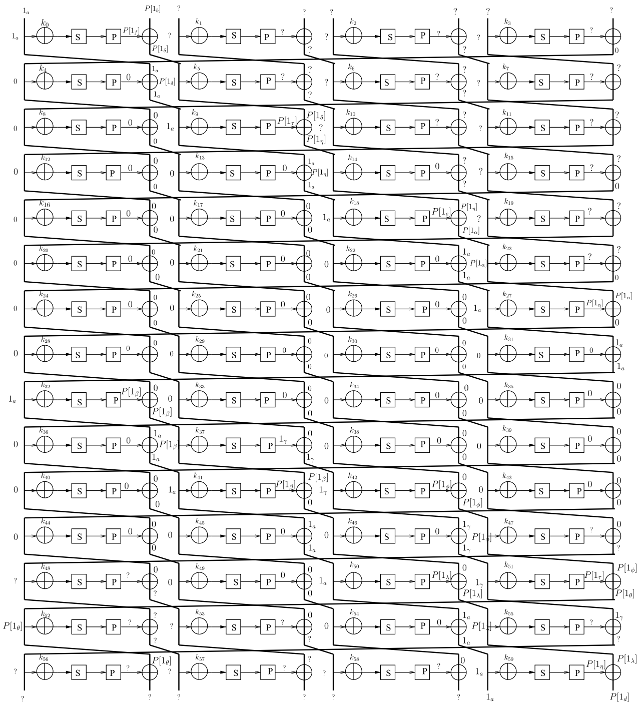

FIGURE 17. **Differential path of fifteen round distinguisher.**

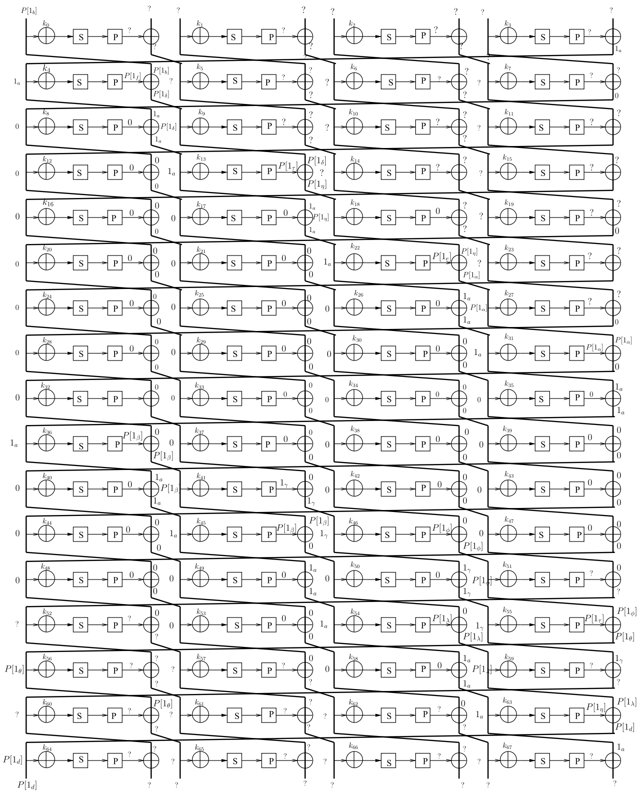

FIGURE 18. **Differential path of seventeen round distinguisher.**

Similar to the backward outbound phase, after getting any such message pair which satisfies the inbound phase, i.e., one starting point, the truncated differential path follows the forward outbound phase with probability 1.

TABLE 5. **Complexity comparison between general attack and proposed distinguisher.**

N	n	c	r	complexity for general attack	complexity of our attack	
					same S-box	different S-box
64	8	4	2	$68 \times r \times 2^2 \approx 2^9$	2^8	2^9
128	16	4	4	$68 \times r \times 2^6 \approx 2^{14}$	2^8	2^{10}
256	32	8	4	$68 \times r \times 2^{12} \approx 2^{21}$	2^{16}	2^{18}

A. SUMMARY OF THE ATTACK

As shown in Table 5, the complexity of the proposed distinguisher is lower than a general attack.

We have presented the complexity of our attack in terms of the number of S-box look ups, as was done in the previous cases. The complexity of our proposed 17-round distinguisher for 8-branch single-SP function is $r \times 2^{2c}$ in time and memory. For a general attack, the required complexity is $68 \times r \times 2^{(\frac{n-c}{2})}$ S-box look ups. Hence the proposed distinguisher has lower complexity than general attack.

VI. CONCLUSIONS

In this work, we have presented 3 distinguishers for two different types of GFN. Our first distinguisher improves the results of Sasaki [20] by extending the attack by one more round.

In our second work we presented two distinguisher for 8-branch type-2 GFN with single-SP function. Our first distinguisher is a 15-round distinguisher for 8-branch type-2 GFN with single-SP functions. The complexity of our proposed distinguisher is much lower than the general attack and it can be used for all the practical design parameters. The complexity of our 17-round distinguisher is the same as the 15-round distinguisher, but the advantage of the distinguisher against a random function is smaller. Further, our 17-round distinguisher works for most practical design parameters for such GFN's.

However, a limitation of all the distinguishers presented in this work is that they are useful only if the input size of S-boxes is greater than or equal to the number of S-boxes used. Removing this limitation and increasing the numbers of rounds attacked for different types of GFN's are interesting open problems.

REFERENCES

[1] M. Luby and C. Rackoff, "How to construct pseudo-random permutations from pseudo-random functions (Abstract)," in *Advances in Cryptology—CRYPTO* (Lecture Notes in Computer Science), vol. 218, H. C. Williams, Ed. Springer, Aug. 1985, p. 447.

[2] T. Shirai, K. Shibutani, T. Akishita, S. Moriai, and T. Iwata, "The 128-bit blockcipher CLEFIA (Extended Abstract)," in *Fast Software Encryption* (Lecture Notes in Computer Science), vol. 218, A. Biryukov, Ed. Springer, Mar. 2007, pp. 181–195.

[3] R. L. Rivest, M. J. B. Robshaw, and Y. L. Yin, "RC6 as the AES," in *Proc. AES Candidate Conf.*, Jan. 2000, pp. 337–342.

[4] D. Hong *et al.*, "Hight: A new block cipher suitable for low-resource device," in *Cryptographic Hardware and Embedded Systems* (Lecture Notes in Computer Science), vol. 4249, L. Goubin and M. Matsui, Eds. Springer, 2006, pp. 46–59.

[5] L. R. Knudsen and V. Rijmen, "Known-key distinguishers for some block ciphers," in *Proc. ASIACRYPT*, Dec. 2007, pp. 315–324.

[6] M. Minier, R. C.-W. Phan, and B. Pousse, "Distinguishers for ciphers and known key attack against rijndael with large blocks," in *Progress in Cryptology—AFRICACRYPT* (Lecture Notes in Computer Science), vol. 5580, B. Preneel, Ed. Springer, 2009, pp. 60–76.

[7] Y. Sasaki, "Known-key attacks on rijndael with large blocks and strengthening *ShiftRow* parameter," in *Advances in Information and Computer Security* (Lecture Notes in Computer Science), vol. 6434, I. Echizen, N. Kunihiro, and R. Sasaki, Eds. Springer, 2010, pp. 301–315.

[8] H. Gilbert and T. Peyrin, "Super-sbox cryptanalysis: Improved attacks for aes-like permutations," in *Fast Software Encryption* (Lecture Notes in Computer Science), vol. 6147, S. Hong and T. Iwata, Eds. Springer, 2010, pp. 365–383.

[9] M. Minier, M. Naya-Plasencia, and T. Peyrin, "Analysis of reduced-shavite-3-256 v2," in *Fast Software Encryption* (Lecture Notes in Computer Science), vol. 6733, A. Joux, Ed. Springer, 2011, pp. 68–87.

[10] H. Kang, D. Hong, D. Moon, D. Kwon, J. Sung, and S. Hong, "Known-key attacks on generalized feistel schemes with sp round function," *IEICE Trans.*, vol. E95-A, no. 9, pp. 1550–1560, Sep. 2012.

[11] Y. Sasaki, S. Emami, D. Hong, and A. Kumar, "Improved known-key distinguishers on feistel-sp ciphers and application to camellia," in *Advances in Information and Computer Security* (Lecture Notes in Computer Science), vol. 7372, W. Susilo, Y. Mu, and J. Seberry, Eds. Springer, 2012, pp. 87–100.

[12] M. Lamberger, F. Mendel, M. Schlaffer, C. Rechberger, and V. Rijmen, "The rebound attack and subspace distinguishers: Application to whirlpool," *J. Cryptol.*, pp. 1–40, Nov. 2013.

[13] E. Biham and A. Shamir, "Differential cryptanalysis of DES-like cryptosystems," in *Advances in Cryptology—CRYPTO* (Lecture Notes in Computer Science), A. Menezes and S. A. Vanstone, Eds. Springer, 1990, pp. 2–21.

[14] M. Matsui, "Linear cryptoanalysis method for DES cipher," in *Proc. EUROCRYPT*, May 1993, pp. 386–397.

[15] A. Bogdanov and K. Shibutani, "Double sp-functions: Enhanced generalized feistel networks—Extended abstract," in *Advances in Information and Computer Security* (Lecture Notes in Computer Science), U. Parampalli and P. Hawkes, Eds. Springer, 2011, pp. 106–119.

[16] W. Wu, W. Zhang, and D. Lin, "Security on generalized feistel scheme with sp round function," *IJ. Netw. Secur.*, vol. 3, no. 3, pp. 215–224, Nov. 2006.

[17] K. Shibutani, "On the diffusion of generalized feistel structures regarding differential and linear cryptanalysis," in *Selected Areas in Cryptography* (Lecture Notes in Computer Science), vol. 6544, A. Biryukov, G. Gong, and D. R. Stinson, Eds. Springer, 2010, pp. 211–228.

[18] Y. Sasaki and K. Yasuda, "Known-key distinguishers on 11-round feistel and collision attacks on its hashing modes," in *Fast Software Encryption* (Lecture Notes in Computer Science), vol. 6733, A. Joux, Ed. Springer, 2011, pp. 397–415.

[19] F. Mendel, C. Rechberger, and M. Schlaffer, and S. S. Thomsen, "The rebound attack: Cryptanalysis of reduced whirlpool and Grostl," in *Fast Software Encryption* (Lecture Notes in Computer Science), vol. 5665, O. Dunkelman, Ed. Springer, 2009, pp. 260–276.

[20] Y. Sasaki, "Double-SP is weaker than single-SP: Rebound attacks on feistel ciphers with several rounds," in *Progress in Cryptology—INDOCRYPT* (Lecture Notes in Computer Science), vol. 7668, S. D. Galbraith and M. Nandi, Eds. Springer, 2012, pp. 265–282.

[21] D. Chang, A. Kumar, and S. K. Sanadhya, "Security analysis of GFN: 8-Round distinguisher for 4-branch type-2 GFN," in *Proc. 14th Int. Conf. Cryptol. (INDOCRYPT)*, Mumbai, India, Dec. 2013, pp. 136–148.

[22] Y. Zheng, T. Matsumoto, and H. Imai, "On the construction of block ciphers: Provably secure and not relying on any unproved hypotheses," in *Advances in Cryptology—CRYPTO* (Lecture Notes in Computer Science), vol. 435, G. Brassard, Ed. Springer, 1989, pp. 461–480.

[23] A. Joux, in *Proc. 18th Int. Workshop Fast Softw. Encryption (FSE)*, vol. 6733. Feb. 2011.

DONGHOON CHANG received the bachelor's degree in mathematics, the master's degree in information security and cryptography, and the Ph.D. degree in information security and cryptography from Korea University, South Korea, in 2001, 2003, and 2008, respectively. He was a Researcher with the University of Waterloo, Canada, in 2006. He held a post-doctoral position at Columbia University, USA, from 2008 to 2009. He was a Researcher with the Computer Security Division, National Institute of Standards and Technology, USA, from 2009 to 2012. Since 2012, he has been an Assistant Professor with the Indraprastha Institute of Information Technology Delhi, India.

ABHISHEK KUMAR received the B.Tech. degree in computer science engineering from the West Bengal University of Technology, India, in 2010, and the M.Tech. degree from the Indraprastha Institute of Information Technology Delhi, New Delhi, India, in 2013, where he is currently pursuing the Ph.D. degree. He is supervised by Dr. S. K. Sanadhya and Dr. D. Chang. His current research interests include design and analysis of block ciphers.

SOMITRA KUMAR SANADHYA received the B.Tech. degree from IIT Delhi in 1994, the M.Tech. degree from Jawaharlal Nehru University, Delhi, in 2002, and the Ph.D. degree from the Indian Statistical Institute, Kolkata, in 2009. He has been an Assistant Professor with Indraprastha Institute of Information Technology Delhi since 2010. His primary research area is cryptology and he was with many Indian governmental agencies for development and analysis of cryptograhic primitives. He has been a Program Chair and a Program Committee Member of many international conferences.