New Linear Cryptanalytic Results of Reduced-Round of CAST-128 and CAST-256*

Meiqin Wang, Xiaoyun Wang, and Changhui Hu

Key Laboratory of Cryptologic Technology and Information Security,
Ministry of Education, Shandong University,
Jinan, 250100, China
mqwang@sdu.edu.cn, xywang@sdu.edu.cn, huchanghui@mail.sdu.edu.cn

Abstract. This paper presents a linear cryptanalysis for reduced round variants of CAST-128 and CAST-256 block ciphers. Compared with the linear relation of round function with the bias 2^{-17} by J. Nakahara et al., we found the more heavily biased linear approximations for 3 round functions and the highest one is $2^{-12.91}$. We can mount the known-plaintext attack on 6-round CAST-128 and the ciphertext-only attack on 4-round CAST-128. Moreover the known-plaintext attack on 24-round CAST-256 with key size 192 and 256 bits has been given, and the ciphertext-only attack on 21-round CAST-256 with key size 192 and 256 bits can be performed. At the same time, we also present the attack on 18-round CAST-256 with key size 128 bits.

Keywords: Linear Cryptanalysis, Block Cipher, CAST-128, CAST-256.

1 Introduction

CAST-128 is a block cipher designed by C. Adams and S. Tavares in 1996[1], and is used in a number of products notably as the default cipher in some versions of GPG and PGP[2,3]. It has been approved for Canadian government use by the Communications Security Establishment. CAST-256 is one of the fifteen candidate algorithms of the first AES Candidate Conference[4,5].

One way to reduce the size of the largest entry in the XOR table is to use injective substitution layer(S-boxes) such that the number of output bits from the S-box is sufficiently larger than the number of input bits. In this way, it is very likely that the entries in the XOR distribution table of a randomly chosen injective S-box will have only small values, making the block cipher resistant to differential cryptanalysis.

In order to resist to differential cryptanalysis, CAST-128 and CAST-256 use injective substitution S-boxes with 32-bit output and 8-bit input. Moreover, S-boxes are designed from bent functions to resist linear cryptanalysis. Therefore,

* Supported by 973 Program No. 2007CB807902, National Natural Science Foundation of China Key Project No. 90604036, National Outstanding Young Scientist No. 60525201.

R. Avanzi, L. Keliher, and F. Sica (Eds.): SAC 2008, LNCS 5381, pp. 429–441, 2009.

the cryptanalysis for them will be very difficult. As far as we know, the differential cryptanalysis of 9 quad-rounds CAST-256 and 5-round CAST-128 under weak-key assumption and the impossible differential cryptanalysis for 20-round CAST-256 have been given respectively in [6] and [7]. In addition, Wagner presented the boomerang attack on 16-round CAST-256[11].

Nakahara and Rasmussen presented the first concrete linear cryptanalysis on reduced-round CAST-128 and CAST-256. They can recover the subkey for 4-round CAST-128 with 2^{37} known plaintexts and $2^{72.5}$ times of 4-round CAST-128 encryption. The distinguishing attack for 12-round CAST-256 with 2^{101} known plaintexts and 2^{101} times of 12-round CAST-256 encryption has been given[8].

In this paper, we give the linear cryptanalysis for 6-round CAST-128 with $2^{53.96}$ known plaintexts and $2^{88.51}$ times of 6-round CAST-128 encryption, and give the linear cryptanalysis for 24-round CAST-256 with $2^{124.10}$ known plaintexts and $2^{156.20}$ times of 24-round CAST-256 encryption. Moreover, we present the ciphertext-only attack on 4-round CAST-128 and 21-round CAST-256.

The paper is organized as follows. Section 2 introduces the description of CAST-128 and CAST-256. In Section 3, we present how to find the more heavily biased linear approximations of three round functions in these two block ciphers. In Section 4, we give the linear cryptanalysis for reduced-round CAST-128. In Section 5, we give the linear cryptanalysis for reduced-round CAST-256. In Section 6, we conclude this paper.

2 Description of CAST-128 and CAST-256

2.1 Description of CAST-128

As a Feistel block cipher, CAST-128 uses a block size 64 bits, and the key size can vary from 40 bits to 128 bits, in 8-bit increments. For key sizes up to and including 80 bits, the number of round is 12. For key sizes greater than 80 bits, the cipher uses the full 16 rounds[1]. The overall operation of CAST-128 is similar to DES[9], which is described in Fig.1. CAST-128 splits the plaintext into left and right 32-bit halves L_0 and R_0. In the key schedule process, 16 pairs of subkeys K_{mi} and K_{ri} for the user key K are computed, with one pair of subkeys per round. A 32-bit key-dependent value K_{mi} is used as a "masking" key and a 5-bit K_{ri} is used as a "rotation" key of the i^{th} round. Our cryptanalysis is not related to the key schedule, so we don't present it in detail. The encryption process is defined as follows,

– For $1 \leq i \leq 16$, compute L_i and R_i as follows:

$$L_i = R_{i-1}$$
$$R_i = L_{i-1} \oplus F_i(R_{i-1}, K_{mi}, K_{ri})$$

where F_i is the round function(F_i is of Type 1, Type 2, or Type 3) described later.

– The ciphertext is (R_{16}, L_{16}).

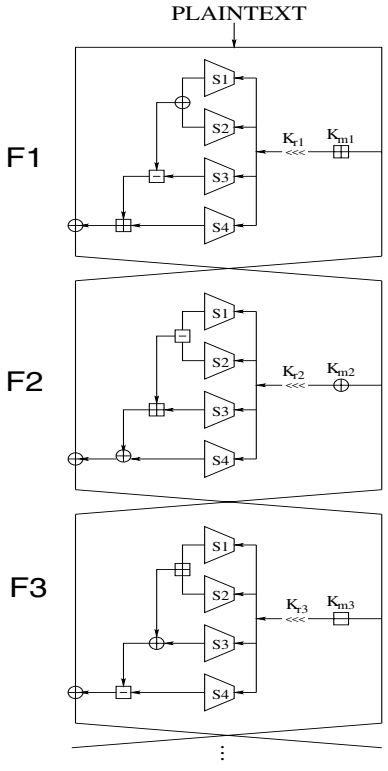

Fig. 1. CAST-128 encryption algorithm

Decryption is identical to the encryption algorithm given above, except that the subkey pairs are used in reverse order to compute (L_0,R_0) from (R_{16},L_{16}).

Three different round functions are used in CAST-128. X is the input to the round function and I is the input to 4 S-boxes where I_a and I_d are the most significant byte and the least significant byte of I respectively($I = I_a \| I_b \| I_c \| I_d$). "+" and "−" are addition and subtraction modulo 2^{32}. "⊕" is bitwise XOR, and "<<<" is the circular left-shift operation. The round functions are defined as follows,

$$
\begin{aligned}
Type1 : I &= ((K_{mi} + X) <<< K_{ri}) \\
F_1 &= ((S_1[I_a] \oplus S_2[I_b]) - S_3[I_c]) + S_4[I_d] \\
Type2 : I &= ((K_{mi} \oplus X) <<< K_{ri}) \\
F_2 &= ((S_1[I_a] - S_2[I_b]) + S_3[I_c]) \oplus S_4[I_d] \\
Type3 : I &= ((K_{mi} - X) <<< K_{ri}) \\
F_3 &= ((S_1[I_a] + S_2[I_b]) \oplus S_3[I_c]) - S_4[I_d]
\end{aligned}
$$

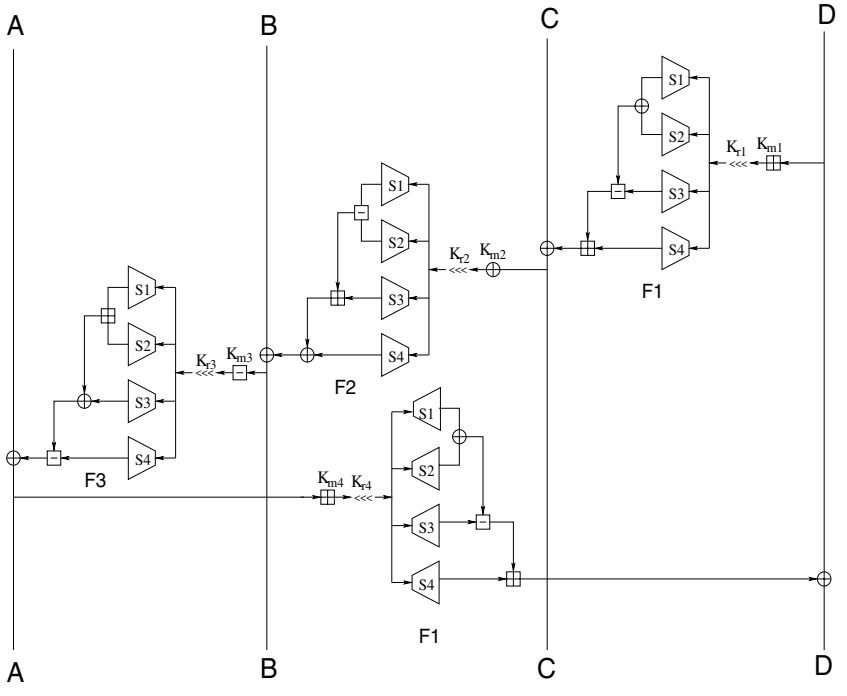

Fig. 2. CAST-256 encryption algorithm

Rounds 1, 4, 7, 10, 13, and 16 use F_1 function. Rounds 2, 5, 8, 11, and 14 use F_2 function. Rounds 3, 6, 9, 12, and 15 use F_3 function. In the above equations, S_1, S_2, S_3, and S_4 are 4 S-boxes, which input is 8-bit and output is 32-bit.

2.2 Description of CAST-256

As a candidate for the first AES conference, CAST-256 is designed based on CAST-128. The block size is 128-bit, and the key size can be 128-bit, 192-bit and 256-bit. The round number is 48 for all key size. The structure for CAST-256 is generalized Feistel Network structure in Fig. 2.

We denote 128-bit block as $\beta = (ABCD)$ where A,B,C and D are each 32 bits in length. Two types of round function, the "forward quad-round" $Q(\cdot)$ and the "reverse quad-round" $\bar{Q}(\cdot)$ are used in CAST-256.

The "forward quad-round" $\beta \longleftarrow Q_i(\beta)$ is defined as the following four rounds,

$$
\begin{aligned}
C &= C \oplus F_1(D, {K_{r1}}^{(i)}, {K_{m1}}^{(i)}) \\
B &= B \oplus F_2(C, {K_{r2}}^{(i)}, {K_{m2}}^{(i)}) \\
A &= A \oplus F_3(B, {K_{r3}}^{(i)}, {K_{m3}}^{(i)}) \\
D &= D \oplus F_1(A, {K_{r4}}^{(i)}, {K_{m4}}^{(i)})
\end{aligned}
$$

And the "reverse quad-round"$\beta \longleftarrow \bar{Q}_i(\beta)$ is defined as the following four rounds,

$$D = D \oplus F_1(A, {K_{r4}}^{(i)}, {K_{m4}}^{(i)})$$
$$A = A \oplus F_3(B, {K_{r3}}^{(i)}, {K_{m3}}^{(i)})$$
$$B = B \oplus F_2(C, {K_{r2}}^{(i)}, {K_{m2}}^{(i)})$$
$$C = C \oplus F_1(D, {K_{r1}}^{(i)}, {K_{m1}}^{(i)})$$

where ${K_r}^{(i)} = \{{K_{r1}}^{(i)}, {K_{r2}}^{(i)}, {K_{r3}}^{(i)}, {K_{r4}}^{(i)}\}$ is the set of rotation keys for the i^{th} quad-round, and ${K_m}^{(i)} = \{{K_{m1}}^{(i)}, {K_{m2}}^{(i)}, {K_{m3}}^{(i)}, {K_{m4}}^{(i)}\}$ is the set of masking keys for the i^{th} quad-round.

The encryption process for CAST-256 consists of 6 "forward quad-rounds" followed by 6 "reverse quad-rounds". Decryption is identical to encryption except that the sets of quad-round keys ${K_r}^{(i)}$ and ${K_m}^{(i)}$ are used in reverse order.

3 Linear Approximation for Round Functions

The S-boxes of CAST-128 have dimension 8×32 bits and are non-surjective, so their linear approximation tables are difficult to be constructed. The probability of the linear approximations for these S-boxes with the form $0 \rightarrow \Gamma$ is away from $\frac{1}{2}$ because of the non-surjective property of S-boxes, where '0' stands for a zero 8-bit mask, and 'Γ' stands for a nonzero 32-bit mask. This kind of linear approximation only represents that an exclusive-or of output bits selected by Γ is zero. Especially if there is only one non-zero bit for Γ, the probability is always equal to $\frac{1}{2} \pm \frac{1}{2^5}$. In [8], in order to obtain the linear approximation for the round function, only the linear approximation for S-boxes with the form $0 \rightarrow 1$ has been used where only the least significant output masking bit is non-zero. Then the bias for the linear approximation of the round function with the form $0 \rightarrow 1$ in Fig.3 is 2^{-17} according to the Piling-Up lemma[10] because the least significant output masking bit is not affected by the mixture operations with modular addition, modular subtraction and XOR operations. In [8], authors think the highest bias for the round function is $0 \rightarrow 1$ because the carry bits in modular addition and the borrow bits in modular subtraction of round function will reduce the bias to less than 2^{-17}, so they use the linear relations for round functions F_1, F_2 or F_3 having the following forms,

$$F_i : 00000000_X \rightarrow 00000000_X$$

$$F_i : 00000000_X \rightarrow 00000001_X$$

Based on the above line relations, 2 types of 2-round iterative linear relations for CAST-128 depicted in Fig.4(a) and Fig.4(b) respectively have been given. According to the Piling-Up lemma[10], the biases for the two 2-round iterative linear relations are all 2^{-17}[8].

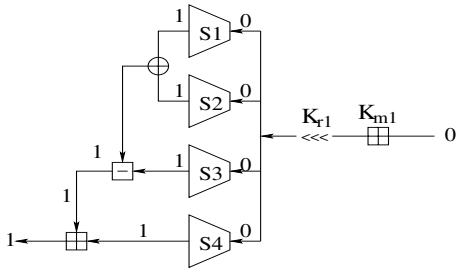

Fig. 3. Bit masks of a linear relation for round function F_1

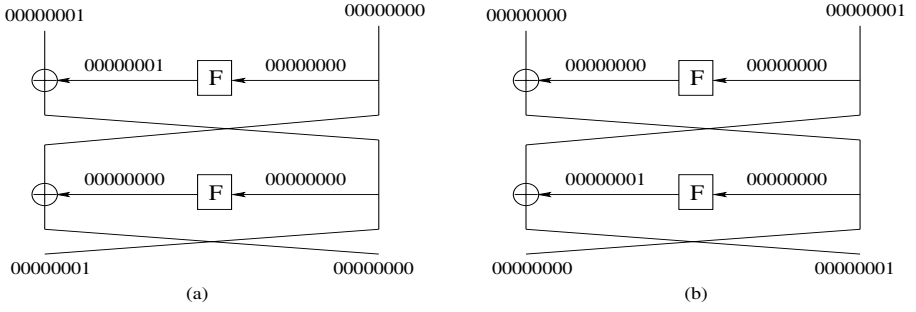

Fig. 4. 2 two-round iterative linear relations for CAST-128

However, we find an important fact that the carry-bit in the modular addition and the borrow-bit in the modular subtraction don't always decrease the bias of linear approximation, sometimes they can further increase the bias. The crypt-analysis in [8] only uses the bias for the single output bit(the least significant bit) of S-boxes. In fact, we find that the non-random properties of the consecutive output bits of S-boxes may result in the higher bias of the output bit of round function with modular addition, modular subtraction and XOR operations compared with the bias of S-boxes output. For example, two least significant bits of S-box output have 4 possible values such as '00', '01', '10' and '11'. If the distribution for the 4 values are non-random(the probabilities are not equal), the bias of the second least-significant bit of round function may be increased after the mixture operations on them. So we searched the linear approximations for the round functions F_1, F_2 and F_3 which have the form $0 \rightarrow \Gamma$ and only one non-zero bit mask of Γ, and the bias for this kind of linear approximation represents the unbalance property for each output bit of round function. The results are presented in Table1. From Table1, we identified the highest bias is not for linear approximation $0 \rightarrow 1$, but the highest biases for F_1, F_2 and F_3 are $2^{-13.71}$, $2^{-14.41}$ and $2^{-14.26}$ respectively which are corresponding to the linear approximation $0 \rightarrow 00000010_X$, $0 \rightarrow 00020000_X$, and $0 \rightarrow 00000080_X$.

Table 1. Linear approximation table for one non-zero bit mask of Γ

non-zero masking bit for Γ	$bias_{F_1} = \lvert P_r - \frac{1}{2} \rvert$	$bias_{F_2} = \lvert P_r - \frac{1}{2} \rvert$	$bias_{F_3} = \lvert P_r - \frac{1}{2} \rvert$
1	$2^{-17.00}$	$2^{-17.00}$	$2^{-17.00}$
2	$2^{-18.00}$	$2^{-17.68}$	$2^{-17.48}$
3	$2^{-18.99}$	$2^{-19.91}$	$2^{-14.48}$
4	$2^{-14.58}$	$2^{-15.00}$	$2^{-15.38}$
5	$2^{-13.98}$	$2^{-14.61}$	$2^{-15.23}$
6	$2^{-13.71}$	$2^{-16.45}$	$2^{-15.54}$
7	$2^{-16.30}$	$2^{-17.00}$	$2^{-17.81}$
8	$2^{-16.91}$	$2^{-18.79}$	$2^{-14.26}$
9	$2^{-15.24}$	$2^{-15.68}$	$2^{-18.20}$
10	$2^{-17.69}$	$2^{-18.47}$	$2^{-17.03}$
11	$2^{-17.38}$	$2^{-18.74}$	$2^{-16.60}$
12	$2^{-15.88}$	$2^{-23.68}$	$2^{-15.41}$
13	$2^{-16.08}$	$2^{-16.38}$	$2^{-16.71}$
14	$2^{-15.69}$	$2^{-14.74}$	$2^{-15.68}$
15	$2^{-17.08}$	$2^{-17.00}$	$2^{-16.80}$
16	$2^{-17.53}$	$2^{-15.19}$	$2^{-19.09}$
17	$2^{-21.54}$	$2^{-17.34}$	$2^{-16.26}$
18	$2^{-14.41}$	$2^{-14.41}$	$2^{-14.47}$
19	$2^{-15.55}$	$2^{-19.30}$	$2^{-17.43}$
20	$2^{-18.96}$	$2^{-15.88}$	$2^{-16.41}$
21	$2^{-17.66}$	$2^{-16.30}$	$2^{-20.80}$
22	$2^{-15.32}$	$2^{-16.80}$	$2^{-19.44}$
23	$2^{-17.20}$	$2^{-15.38}$	$2^{-16.17}$
24	$2^{-18.47}$	$2^{-17.93}$	$2^{-18.73}$
25	$2^{-17.23}$	$2^{-17.64}$	$2^{-15.74}$
26	$2^{-15.77}$	$2^{-16.75}$	$2^{-15.37}$
27	$2^{-14.72}$	$2^{-16.19}$	$2^{-16.44}$
28	$2^{-17.60}$	$2^{-20.46}$	$2^{-17.33}$
29	$2^{-20.12}$	$2^{-17.85}$	$2^{-17.64}$
30	$2^{-16.06}$	$2^{-15.31}$	$2^{-16.34}$
31	$2^{-16.24}$	$2^{-16.23}$	$2^{-18.09}$
32	$2^{-15.82}$	$2^{-16.03}$	$2^{-16.89}$

Additionally, the unbalance property of the single output bit of round function will result in the heavily biased linear approximation with more non-zero output masking bits. So we searched the linear approximations for 3 round functions which have the form $0 \rightarrow \Gamma$ with two and three non-zero masking bits of Γ. Further four and five non-zero masking bits of Γ for F_2 have been examined, but we have not examined four or five non-zero masking bits of Γ for F_1 and F_3 and more than five non-zero masking bits for 3 round functions because the complexity of computation is very large. Their linear relations with the highest bias we have found will be given in Table 2.

From Table 1 and Table 2, the best bias for single round function we found is $2^{-12.91}$ corresponding to the linear relation $00000000_X \rightarrow 03400000_X$ for F_2.

Table 2. Best linear approximation for more non-zero bits of Γ

Function Type	Γ	Number of non-zero bits of Γ	$bias = \lvert P_r - \frac{1}{2} \rvert$
F_1	$0000000C_X$	2	$2^{-14.07}$
F_2	80004000_X	2	$2^{-13.06}$
F_3	02400000_X	2	$2^{-13.71}$
F_1	02600000_X	3	$2^{-13.37}$
F_2	03400000_X	3	$2^{-12.91}$
F_3	00030020_X	3	$2^{-14.05}$
F_2	00600300_X	4	$2^{-13.64}$
F_2	32000900_X	5	$2^{-13.48}$

4 Linear Cryptanalysis for Reduced-Round CAST-128

4.1 Known-Plaintext Attack for Reduced-Round CAST-128

Based on the above linear approximations of the 3 round functions, we can obtain the 5-round linear relation in Fig 5.a. The output mask Γ in round 2 and round 4 is non-zero, but zero in round 1, 3 and 5. The input mask from the first round to the fifth round are all zero. So the probability of the linear relation in round 1, 3 and 5 are all 1. The bias of the linear relation $00000000_X \rightarrow 03400000_X$ for F_1 is $2^{-13.57}$, and the bias of the linear relation $00000000_X \rightarrow 03400000_X$ for F_2 is $2^{-12.91}$. Based on "the Piling-Up lemma", the bias for the 5-round linear approximation is $2^{-25.48}$.

The linear relation in Fig 5.a is a 5-round distinguisher from the random permutation, which can be presented as follows,

$$(P_R \oplus C_R) \cdot 03400000_X = 0$$

where P_R is the right 32-bit of the plaintext, and C_R is the right 32-bit of the ciphertext for 5-round. As a known plaintext attack, the number of known plaintext N required in linear cryptanalysis is proportional to ϵ^{-2}[10], where ϵ is the bias for the linear relation. If N is taken as $8 \cdot \epsilon^{-2}$, the attack will be successful with very high probability. So we can distinguish 5-round CAST-128 with $8 \cdot 2^{25.48 \cdot 2} = 2^{53.96}$ known plaintexts.

We can recover 37-bit subkey of 6-round using the above 5-round distinguisher in Fig 5.a. As the distinguishing attack for 5-round, the attack also requires $2^{53.96}$ known plaintexts and $2^{53.96} \cdot 2^{37} = 2^{90.96}$ one-round encryptions, which is equivalent to $2^{88.51}$ 6-round encryptions.

4.2 Ciphertext-Only Attack for Reduced-Round CAST-128

If the plaintext is ASCII encoded English text, we can attack reduced-round CAST-128 only with ciphertexts. We use the linear approximation for 3-round CAST-128 where only F_2 is active,

$$(P_R \oplus R_3) \cdot 00008000_X = 0$$

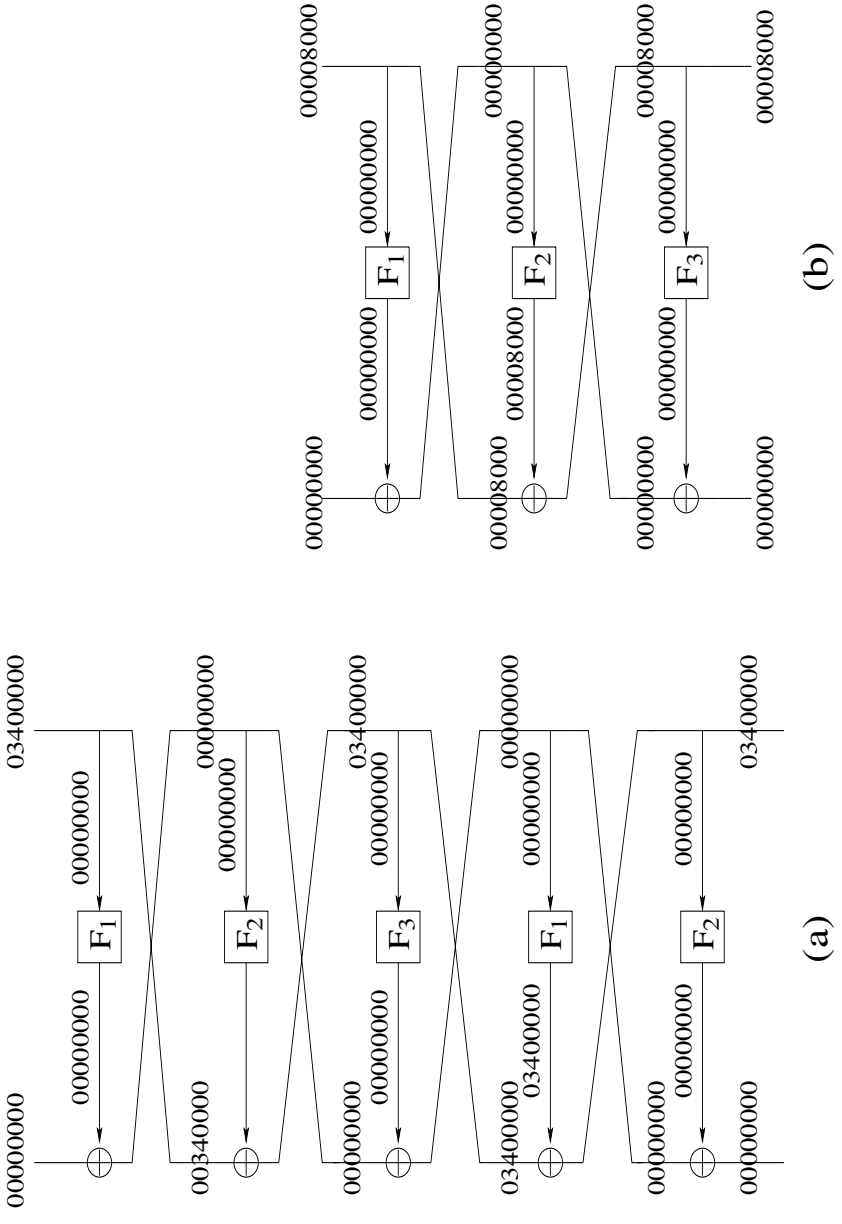

Fig. 5. Two linear relations for CAST-128

where R_3 is the right 32-bit output for round 3, and the bias for the above linear approximation is $2^{-15.19}$, so we can construct the distinguisher of 3-round CAST-128 with only $8 \cdot 2^{15.19 \cdot 2} = 2^{33.38}$ ciphertexts in Fig 5.b. Moreover we can recover 37-bit subkey of 4-round using the above 3-round distinguisher. The

attack also requires only $2^{33.38}$ ciphertexts and $2^{33.38} \cdot 2^{37} = 2^{70.38}$ one-round encryptions, which is equivalent to $2^{68.38}$ 4-round encryptions.

5 Linear Cryptanalysis for Reduced-Round CAST-256

5.1 Known-Plaintext Attack for Reduced-Round CAST-256

As described in Section 3, the highest bias for single round function we found is $2^{-12.91}$ corresponding to the linear relation $0 \rightarrow 03400000_X$ for F_2. So we arrive the iterative linear approximation for one quad-round CAST-256 in Fig6.a. Only F_2 in each quad-round is active, but other 3 round functions are all non-active. We can derive the linear approximation for r quad-rounds of CAST-256 which can be used as a distinguisher, which can be represented as follows,

$$(B \oplus F) \cdot 03400000_X = 0$$

where (A, B, C, D) and (E, F, G, H) denote the plaintext block and the ciphertext block for r quad-rounds respectively. Based on "the Piling-Up lemma", the bias for the linear approximation is $2^{r-1} \cdot 2^{-12.91 \cdot r}$.

We can distinguish 21 rounds CAST-256 from a random permutation with $2^{124.1}$ known plaintexts. By the 21 rounds distinguisher, we can recover 37-bit subkey of round 22 for 24-round CAST-256 with the key size 192 or 256 bits. The time complexity is $2^{124.1} \cdot 2^{37} = 2^{161.1}$ one-round CAST-256 encryptions which is equivalent to $2^{156.2}$ 24-round CAST-256 encryptions.

For CAST-256 with key size 128 bits, we use the linear approximation $0 \rightarrow 02600000_X$ for F_1 with the bias $2^{-13.37}$ to construct the iterative quad-round linear approximation in Fig 6.b. So the iterative linear approximation for 3 quad-round CAST-256 can be derived. Only F_1 of the 4^{th} round in each quad-round is active, but other 3 round functions are all non-active. The bias for the linear approximation is $2^{-38.11}$ and we can recover 37-bit subkey of round 16 with $2^{79.22}$ known plaintexts and $2^{111.98}$ times of 18-round CAST-256 encryption.

5.2 Ciphertext-Only Attack for Reduced-Round CAST-256

If the plaintext is ASCII encoded English text, we can attack reduced-round CAST-256 only with ciphertexts. We use the linear approximation $0 \rightarrow 00000080_X$ for round function F_3 with bias $2^{-14.26}$, so we obtain the iterative linear approximation for one quad-round CAST-256 in Fig6.c. Only F_3 in round-3 is active, but other 3 round functions are all non-active. We can derive the linear approximation for r quad-rounds of CAST-256 which can be used as a distinguisher, which can be represented as follows,

$$(A \oplus E) \cdot 00000080_X = 0$$

where (A, B, C, D) and (E, F, G, H) denote the plaintext block and the ciphertext block for r quad-rounds respectively. Based on "the Piling-Up lemma", the bias for the linear approximation is $2^{r-1} \cdot 2^{-14.26 \cdot r}$.

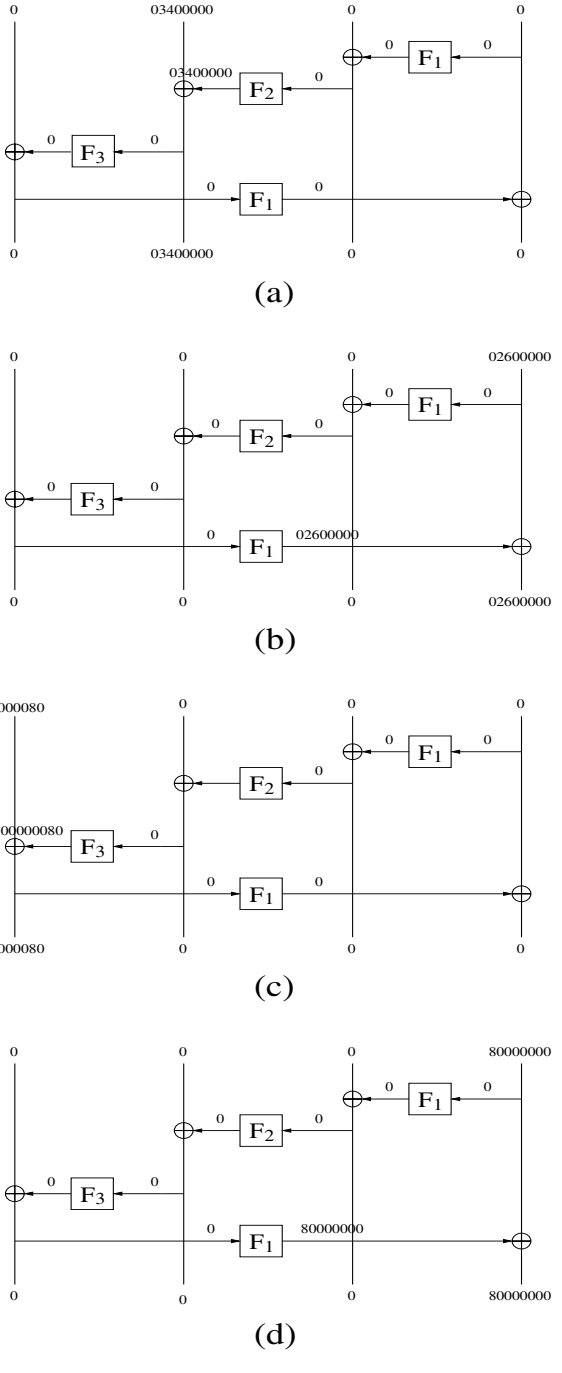

Fig. 6. One quad-round iterative linear relation for CAST-256

We can distinguish 4 quad-rounds CAST-256 from a random permutation with only $2^{111.08}$ ciphertexts. Using 4 quad-rounds distinguisher with only $2^{111.08}$ ciphertexts, we can recover the round 19 subkey for 21-round CAST-256 with the key size 192 or 256 bits. The time complexity is $2^{111.08} \cdot 2^{37} = 2^{148.08}$ one-round CAST-256 encryptions which is equivalent to $2^{143.50}$ 21-round CAST-256 encryptions.

For CAST-256 with key size 128 bits, we use the linear relation $0 \rightarrow 80000000_X$ for F_1 with the bias $2^{-15.82}$ to construct the iterative linear approximation for a quad-round CAST-256 in Fig6.d. So the iterative linear approximation for 3 quad-rounds CAST-256 can be derived. Only F_1 of the 4^{th} round in each quad-round is active, but other 3 round functions are all non-active. The bias for the linear approximation is $2^{-45.46}$ and we can recover the subkey of round 16 with $2^{93.92}$ only-ciphertexts and $2^{126.28}$ times of 18-round CAST-256 encryption.

6 Summary

In this paper, we found that the unbalance for the consecutive bits from S-boxes output may further increase the unbalance of the output from the round function which performs modular addition, modular subtraction and XOR operations on the outputs of 4 S-boxes, This observation led us to find the heavily biased linear relation for the round functions of CAST-128 and CAST-256. After that, we present the best known linear attack on reduced-round CAST-128 and CAST-256. Our attacks are by far the best known attacks on the two ciphers without weak-key assumption. Moreover we give the first ciphertext only attack for reduced round variants of the two ciphers.

We attack 6-round CAST-128, which works for the key size more than 88 bits, with data complexity of $2^{53.96}$ known plaintexts, the time complexity of $2^{88.51}$ times of 6-round encryption. Moreover we mount a ciphertext-only attack on 4-round CAST-128 for the key size more than 68 bits, and the attack uses only $2^{33.38}$ ciphertexts and $2^{68.38}$ times of 4-round encryption. Then we present an attack on 24-round CAST-256 requiring $2^{124.10}$ known plaintexts, $2^{156.20}$ times of 24-round encryptions. In addition, we mount a ciphertext-only attack on 21-round CAST-256 with only $2^{111.08}$ ciphertexts and $2^{143.50}$ 21-round encryptions.

Table 3. Summary of linear attacks on reduced-round CAST-128

Rounds	Key Size	Data Complexity	Time Complexity	Type	Source
2	all	2^{37} KPs	2^{37}	Distinguishing	[8]
3	all	2^{37} KPs	2^{37}	Distinguishing	[8]
	>72 bits	2^{37} KPs	$2^{72.5}$	Key Recovery	[8] [1]
4	>72 bits	2^{37} KPs	$2^{72.5}$	Key Recovery	[8]
	>68 bits	$2^{33.38}$ COs	$2^{68.38}$	Key Recovery	This Paper
6	>88 bits	$2^{53.96}$ KPs	$2^{88.51}$	Key Recovery	This Paper

KPs:Known Plaintexts, COs:Ciphertexts only

Table 4. Summary of linear attacks on reduced-round CAST-256

Rounds	Key Size	Data Complexity	Time Complexity	Type	Source
9	all	2^{69} KPs	2^{103}	Key Recovery	[8]
12	all	2^{101} KPs	2^{101}	Distinguishing	[8]
18	all	$2^{79.22}$ KPs	$2^{111.98}$	Key Recovery	This Paper
	all	$2^{93.92}$ COs	$2^{126.28}$	Key Recovery	This Paper
21	192-bit or 256-bit	$2^{111.08}$ COs	$2^{143.50}$	Key Recovery	This Paper
24	192-bit or 256-bit	$2^{124.1}$ KPs	$2^{156.20}$	Key Recovery	This Paper

[2] KPs:Known Plaintexts, COs:Ciphertexts only

Table 3 and Table 4 give the comparison of our results with the previous linear attacks on CAST-128 and CAST-256.

References

1. Adams, C., Tavares, S.: The CAST-128 Encryption Algorithm. RFC 2144 (May 1997)
2. GnuPG, Gnu Privacy Guard, `http://www.gnupg.org/(en)/features.html`
3. PGP, Pretty Good Privacy, `http://www.pgp.com/`
4. Adams, C., Gilchrist, J.: The CAST-256 Encryption Algorithm. RFC 2612 (June 1999)
5. First AES Candidate Conference, `http://csrc.nist.gov/archive/aes/round1/conf1/aes1conf.htm`
6. Biham, E.: A Note on Comparing the AES Candidates, The AES Development Process, `http://csrc.nist.gov/archive/aes/round1/conf2/papers/biham2.pdf`
7. Seki., H., Kaneko., T.: Differential Cryptanalysis of CAST-256 Reduced to Nine Quad-rounds. Leice Transactions on Fundamentals of Electronics Communications and Computer Sciences E84A(4), 913–918 (2001)
8. Nakahara Jr., J., Rasmussen, M.: Linear Analysis of Reduced-round CAST-128 and CAST-256, SBSEG2007, pp.45–55 (2007)
9. NBS, Data Encryption Standard (DES), FIPS PUB 46, Federal Information Processing Standards Publication 46, U.S. Department of Commerce (January 1977)
10. Matsui, M.: Linear cryptanalysis method for DES cipher. In: Helleseth, T. (ed.) EUROCRYPT 1993. LNCS, vol. 765, pp. 386–397. Springer, Heidelberg (1994)
11. Wagner, D.: The boomerang attack. In: Knudsen, L.R. (ed.) FSE 1999. LNCS, vol. 1636, p. 156. Springer, Heidelberg (1999)