New Methodologies for Differential-Linear Cryptanalysis and Its Extensions

Jiqiang Lu

Department of Mathematics and Computer Science,
Eindhoven University of Technology
5600 MB, Eindhoven, The Netherlands
lvjiqiang@hotmail.com

Abstract. In 1994 Langford and Hellman introduced differential-linear cryptanalysis, which involves building a differential-linear distinguisher by concatenating a linear approximation with such a (truncated) differential that with probability 1 does not affect the bit(s) concerned by the input mask of the linear approximation. In 2002 Biham, Dunkelman and Keller presented an enhanced approach to include the case when the differential has a probability smaller than 1; and in 2005 they proposed several extensions of differential-linear cryptanalysis, including the high-order differential-linear analysis, the differential-bilinear analysis and the differential-bilinear-boomerang analysis. In this paper, we show that Biham et al.'s methodologies for computing the probabilities of a differential-linear distinguisher, a high-order differential-linear distinguisher, a differential-bilinear distinguisher and a differential-bilinear-boomerang distinguisher do not have the generality to describe the analytic techniques. Thus the previous cryptanalytic results obtained by using these techniques of Biham et al. are questionable. Finally, from a mathematical point we give general methodologies for computing the probabilities. The new methodologies lead to some better cryptanalytic results, for example, differential-linear attacks on 13-round DES and 10-round CTC2 with a 255-bit block size and key.

Key words: Block cipher, DES, CTC2, Serpent, Differential-linear cryptanalysis, High-order differential cryptanalysis, Bilinear cryptanalysis, Boomerang analysis

1 Introduction

Differential cryptanalysis was introduced in 1990 by Biham and Shamir [1]. Linear cryptanalysis was introduced in 1992 by Matsui and Yamagishi [2]. A differential cryptanalysis attack is based on the use of one or more differentials, and a linear cryptanalysis attack is based on the use of one or more linear approximations. Both the cryptanalytic methods were applied to attack the full DES [3] cipher faster than an exhaustive key search [4, 5].

In 1994 Langford and Hellman [6] introduced a combination of differential and linear cryptanalysis, known as differential-linear cryptanalysis, and applied it to break 8-round DES. Different from differential and linear cryptanalysis, differential-linear cryptanalysis treats a block cipher as a cascade of two sub-ciphers, and it uses a linear approximation for a sub-cipher and for the other sub-cipher uses a truncated differential [7] that with probability 1 does not affect the bit(s) concerned by the input mask of the linear approximation. In 2002 Biham, Dunkelman and Keller [8] introduced an enhanced version of differential-linear cryptanalysis, which includes the case when the differential has a smaller probability; and they finally described a differential-linear attack on 9-round DES. In 2005, Biham, Dunkelman and Keller [9] introduced several extensions of differential-linear cryptanalysis, including the high-order differential-linear analysis, the differential-bilinear analysis and the differential-bilinear-boomerang analysis. Differential-linear cryptanalysis has been used to yield the best currently published cryptanalytic results for a number of state-of-the-art block ciphers [8, 10–13].

In this paper, we examine the probabilities of a differential-linear distinguisher, a high-order differential-linear distinguisher, a differential-bilinear distinguisher and a differential-bilinear-boomerang distinguisher, find that Biham et al.'s methodologies for computing the probabilities of the distinguishers are not correct in some situations, and it is likely that they do not hold in the general situation; that is to say, they do not have the generality to describe the analytic techniques. An incorrect methodology may lead to a false probability for a distinguisher, and thus the resulting attack will have an erroneous data and time complexity, break a wrong number of rounds of the concerned cipher, and is even ineffective in extreme situations. As a consequence, the previous cryptanalytic results obtained by using these techniques of Biham et al. are highly questionable. Finally, from a mathematical point we deduce general

Table 1. Comparison of previous and our main differential-linear cryptanalytic results on DES and Summary of main cryptanalytic results on CTC2 and Serpent

Cipher	Key Size	Rounds	Attack Technique	Data	Time	Source	Note
DES	56	8	Differential-linear	768 CP	$2^{14.6}$ Enc.	[6]	
		9		$2^{15.75}$ CP	$2^{29.17}$ Enc.	[14]	†
		12		$2^{50.6}$ CP	$2^{52.34}$ Enc.	This paper	
		13		$2^{52.27}$ CP	$2^{52.97}$ Enc.+$2^{57.27}$ MA	This paper	
CTC2 (a 255-bit block size)	255	6	Algebraic	4 CP	2^{253} Enc.	[15]	
		7	Differential	2^{15} CP	2^{15} Enc.	[13]	
		8	Differential-linear	2^{37} CP	2^{37} Enc.	[13]	†
		10		2^{142} CP	2^{207} Enc.	This paper	
Serpent	128, 192, 256	7	Differential	2^{84} CP	$2^{78.9}$ Enc.	[16]	
		10	Linear	$2^{120.6}$ KP	2^{85} Enc.	[17]	
				$2^{118.6}$ KP	2^{85} Enc.	This paper	
		10	Differential-linear	$2^{101.2}$ CP	$2^{115.2}$ Enc.	[12]	†
				$2^{123.4}$ CP	$2^{123.4}$ Enc.	This paper	
	192, 256	8	Amplified boomerang	2^{114} CP	2^{179} Enc.	[19]	
		10	Boomerang [18]	$2^{126.3}$ ACPC	2^{165} Enc.	[20]	
		10	Rectangle	$2^{126.3}$ CP	2^{165} Enc.	[20]	
		11	Linear	$2^{122.9}$ KP	2^{189} Enc.	[17]	
		11	Differential-linear	$2^{121.8}$ CP	$2^{135.7}$ MA	[12]	†
		11		$2^{125.5}$ CP	$2^{148.1}$ Enc.	This paper	
	256	8	Differential	2^{84} CP	$2^{206.7}$ Enc.	[16]	
		9	Amplified boomerang	2^{110} CP	2^{252} Enc.	[19]	
		12	Differential-linear	$2^{123.5}$ CP	$2^{249.4}$ Enc.	[12]	†
				$2^{125.5}$ CP	$2^{244.9}$ Enc.	This paper	

†: The result is based on Biham et al.'s methodology.

methodologies for computing the probabilities. Using the new methodology we present differential-linear attacks on 13-round DES, 10-round CTC2 with a 255-bit block size and key, and 12-round Serpent with a 256-bit key. Table 1 compares previous with our main differential-linear cryptanalytic results on DES and sumarises both the previous and our main cryptanalytic results on CTC2 and Serpent, where CP, KP and ACPC refer respectively to the required numbers of chosen plaintexts, known plaintexts, and adaptively chosen plaintexts and ciphertexts, Enc. refers to the required number of encryption operations of the relevant version of CTC2, DES and Serpent, and MA refers to the number of memory accesses.

The remainder of the paper is organised as follows. In the next section we briefly describe some notation, differential cryptanalysis and linear cryptanalysis. In Section 3 we give the general methodology for computing the probability of a differential-linear distinguisher, and present our cryptanalytic results on CTC2, DES and Serpent in Sections 4–6. In Sections 7–9, we give the general methodologies for computing the probabilities of the remaining three types of distinguishers, and present corrected versions for some previous cryptanalytic results. Section 10 concludes this paper.

2 Preliminaries

In this section we describe some notation and the basic notions used in differential and linear cryptanalysis.

2.1 Notation

In the following descriptions, we assume that a number without a prefix is in decimal notation, and a number with prefix $0x$ is in hexadecimal notation, unless otherwise stated. The bits of a value are numbered from right to left, except in the case of DES, where we use the same numbering notation as in [3]; the leftmost bit is the most significant bit, and the rightmost bit is the least significant bit. We use the following notation.

$\oplus$	bitwise logical exclusive OR (XOR) of two bit strings of the same length
$\odot$	dot product of two bit strings of the same length
$\|\|$	sting concatenation
$\ll$	left shift of a bit string
$\lll$	left rotation of a bit string
$\circ$	functional composition. When composing functions X and Y, X $\circ$ Y denotes the function obtained by first applying X and then applying Y
e_j	a 255-bit value with zeros everywhere except for bit position j, $(0 \leq j \leq 254)$
$e_{i_0,\cdots,i_j}$	the 255-bit value equal to $e_{i_0} \oplus \cdots \oplus e_{i_j}$, $(0 \leq i_0, \cdots, i_j \leq 254)$
$\overline{e}_{i_0,\cdots,i_j,\sim}$	a 255-bit value that has zeros in bit positions $i_0, \cdots, i_j$, and indeterminate values in the other bit positions, $(0 \leq i_0, \cdots, i_j \leq 254)$
$\mathbb{E}$	an n-bit block cipher with a k-bit key
$\star$	an indeterminate value (two values expressed by the $\star$ symbol may be different)

2.2 Differential Cryptanalysis

Differential cryptanalysis [1] takes advantage of how a specific difference in a pair of inputs of a cipher can affect a difference in the pair of outputs of the cipher, where the pair of outputs are obtained by encrypting the pair of inputs using the same key. The notion of difference can be defined in several ways; the most widely discussed is with respect to the XOR operation. The difference between the inputs is called the input difference, and the difference between the outputs of a function is called the output difference. The combination of the input difference and the output difference is called a differential. The probability of a differential is defined as follows.

Definition 1. *If α and β are n-bit blocks, then the probability of the differential (α, β) for $\mathbb{E}$, written $\mathit{\Delta}\alpha \rightarrow \mathit{\Delta}\beta$, is defined to be*

$$\Pr_{\mathbb{E}}(\mathit{\Delta}\alpha \rightarrow \mathit{\Delta}\beta) = \Pr_{P \in \{0,1\}^n}(\mathbb{E}(P) \oplus \mathbb{E}(P \oplus \alpha) = \beta).$$

For a random function, the expected probability of a differential for any pair (α, β) is 2^{-n}. Therefore, if $\Pr_{\mathbb{E}}(\mathit{\Delta}\alpha \rightarrow \mathit{\Delta}\beta)$ is larger than 2^{-n}, we can use the differential to distinguish $\mathbb{E}$ from a random function, given a sufficient number of chosen plaintext pairs.

2.3 Linear Cryptanalysis

Linear cryptanalysis [2, 5] exploits correlations between a particular linear function of the input blocks and a second linear function of the output blocks. The combination of the two linear functions is called a linear approximation. The most widely used linear function involves computing the bitwise dot product operation of the block with a specific binary vector (the specific value combined with the input blocks may be different from the value applied to the output blocks). The value combined with the input blocks is called the input mask, and the value applied to the output blocks is called the output mask. The probability of a linear approximation is defined as follows.

Definition 2. *If α and β are n-bit blocks, then the probability of the linear approximation (α, β) for $\mathbb{E}$, written $\mathit{\Gamma}\alpha \rightarrow \mathit{\Gamma}\beta$, is defined to be*

$$\Pr_{\mathbb{E}}(\mathit{\Gamma}\alpha \rightarrow \mathit{\Gamma}\beta) = \Pr_{P \in \{0,1\}^n}(P \odot \alpha = \mathbb{E}(P) \odot \beta).$$

We refer to the dot product $P \odot \alpha$ as the input parity, and the dot product $\mathbb{E}(P) \odot \beta$ as the output parity.

For a random function, the expected probability of a linear approximation for any pair (α, β) is $\frac{1}{2}$. The bias of a linear approximation $\mathit{\Gamma}\alpha \rightarrow \mathit{\Gamma}\beta$, denoted by ϵ, is defined to be $\epsilon = |\Pr_{\mathbb{E}}(\mathit{\Gamma}\alpha \rightarrow \mathit{\Gamma}\beta) - \frac{1}{2}|$. Thus, if the bias ϵ is sufficiently large, we can use the linear approximation to distinguish $\mathbb{E}$ from a random function, given a sufficient number of matching plaintext-ciphertext pairs.

3 Methodology for Differential-Linear Cryptanalysis

In this section we first review differential-linear cryptanalysis, then deduce the probability of a differential-linear distinguisher, and discuss its implications.

3.1 Review of Differential-Linear Cryptanalysis

In 1994 Langford and Hellman [6] introduced differential-linear cryptanalysis, which uses a so-called differential-linear distinguisher. To define a differential-linear distinguisher we need to treat $\mathbb{E}$ as a cascade of two sub-ciphers $\mathbb{E}_0$ and $\mathbb{E}_1$, where $\mathbb{E} = \mathbb{E}_0 \circ \mathbb{E}_1$. A differential-linear distinguisher is then defined to be a pair consisting of a (truncated) differential and a linear approximation $(\Delta\alpha \to \Delta\beta, \Gamma\gamma \to \Gamma\delta)$, where $\Gamma\gamma \to \Gamma\delta$ is a linear approximation with bias ϵ for $\mathbb{E}_1$, and $\Delta\alpha \to \Delta\beta$ is a (truncated) differential for $\mathbb{E}_0$ that with probability 1 has a zero output difference in the bit positions concerned by the linear approximation. Given a pair of plaintexts $(P, P^* = P \oplus \alpha)$, we have $\mathbb{E}_0(P) \odot \gamma = \mathbb{E}_0(P^*) \odot \gamma$ with probability 1. The differential-linear distinguisher is concerned with the event $\delta \odot \mathbb{E}(P) = \delta \odot \mathbb{E}(P^*)$, and thus it has a probability $\Pr(\delta \odot \mathbb{E}(P) = \delta \odot \mathbb{E}(P^*)) = (\frac{1}{2} + \epsilon) \times (\frac{1}{2} + \epsilon) + (\frac{1}{2} - \epsilon) \times (\frac{1}{2} - \epsilon) = \frac{1}{2} + 2\epsilon^2$.

By contrast, for a random function, the expected probability of a differential-linear distinguisher is $\frac{1}{2}$. Therefore, if the bias $|\Pr(\delta \odot \mathbb{E}(P) = \delta \odot \mathbb{E}(P^*)) - \frac{1}{2}| = 2\epsilon^2$ is sufficiently large, we can distinguish $\mathbb{E}$ from a random function.

In 2002 Biham, Dunkelman and Keller [8] presented an enhanced approach to include the case when the differential $\Delta\alpha \to \Delta\beta$ meets the condition $\mathbb{E}_0(P) \odot \gamma = \mathbb{E}_0(P^*) \odot \gamma$ with probability p, where p may be smaller than 1.[1] A more detailed description was given in the PhD thesis of Dunkelman [14]. When the differential meets, they applied Langford and Hellman's analysis described above; and for the other cases they assumed a random distribution for the output parities $\delta \odot \mathbb{E}(P)$ and $\delta \odot \mathbb{E}(P^*)$. Finally they got $\Pr(\delta \odot \mathbb{E}(P) = \delta \odot \mathbb{E}(P^*)) = p \times (\frac{1}{2} + 2\epsilon^2) + (1-p) \times \frac{1}{2} = \frac{1}{2} + 2p\epsilon^2$. We note that a different but equivalent assumption is used in other papers of Biham et al., [9] say, which also leads to the same result, where they assumed that $\mathbb{E}_0(P) \odot \gamma = \mathbb{E}_0(P^*) \odot \gamma$ holds with half chance for the other cases, and got $\Pr(\delta \odot \mathbb{E}(P) = \delta \odot \mathbb{E}(P^*)) = [p + (1-p)\frac{1}{2}] \times (\frac{1}{2} + 2\epsilon^2) + [1 - p - (1-p)\frac{1}{2}] \times [(\frac{1}{2} + \epsilon) \times (\frac{1}{2} - \epsilon) + (\frac{1}{2} - \epsilon) \times (\frac{1}{2} + \epsilon)] = \frac{1}{2} + 2p\epsilon^2$.

As a result, if the bias $2p\epsilon^2$ is sufficiently large, the distinguisher can be used as the basis of a differential-linear attack to distinguish $\mathbb{E}$ from a random function. The attack has a data complexity of about $O(p^{-2}\epsilon^{-4})$.

3.2 General Assumptions

Before further proceeding, we make clear some general assumptions behind differential cryptanalysis, linear cryptanalysis and differential-linear cryptanalysis as well as its extensions.

Differential and linear cryptanalyses are statistical cryptanalytic methods. In practice, a multi-round differential (or linear approximation) is usually constructed by concatenating a few one-round differentials (linear approximations) under the assumption that the involved rounds are independent or an assumption with a similar meaning. As mentioned in [21], this is "most often not exactly the case, but as often it is a good approximation". Differential and linear cryptanalyses, differential-linear cryptanalysis and its extensions generally treat a basic unit of input (i.e. a chosen-plaintext pair for differential cryptanalysis, differential-(bi)linear cryptanalysis, and the differential-(bi)linear-boomerang analysis; a known-plaintext for linear cryptanalysis; and a structure of chosen plaintexts for the high-order differential-linear analysis) as a random variable, and assume that given a set of inputs of the basic unit, the inputs that satisfy the required property can be approximated by an independent distribution, as followed in [5, 8, 9, 22]. Differential-linear cryptanalysis treats the two linear approximations for a basic input as independent.

These assumptions mean that, in some cases, the probability of a differential, linear approximation, or distinguisher may be overestimated or underestimated. However, computer experiments have shown that they work well in practice for some block ciphers; see [5, 10, 23, 24] for examples. It seems reasonable to take them for a theoretical approximation. As a result, like the previous work [4–6, 8, 9] we make use of the assumptions to obtain the formulas for computing the probabilities of the distinguishers; otherwise, these formulas could not be so simple, but more accurate versions can be easily obtained from our reasonings, though a little complicated. Anyway, we suggest that if possible an attacker should check the validity of these assumptions when applying them to specific ciphers.

[1] A more general condition is $\mathbb{E}_0(P) \odot \gamma \oplus \mathbb{E}_0(P^*) \odot \gamma = c$, where $c \in \{0, 1\}$ is a constant. Without loss of generality, we consider the case with $c = 0$.

3.3 A Counterexample to Biham et al.'s Methodology

A differential-linear distinguisher plays a fundamental role in a differential-linear cryptanalysis attack. Biham et al.'s enhanced approach [8] aims to make a differential-linear distinguisher cover more rounds of a block cipher, so that an attacker can break more rounds of the cipher. They used a heuristic way to get the formula for computing the probability of a differential-linear distinguisher by assuming that $\mathbb{E}_0(P) \odot \gamma = \mathbb{E}_0(P^*) \odot \gamma$ holds with half chance for the other cases (or its equivalent).

However, we find that their methodology is not correct for some situations; for example, let's intuitively consider the situation where the differential $\Delta\alpha \to \Delta\beta$ meets $\beta \odot \gamma = 0$ with probability $\frac{1}{2}$, and all the other possible differentials $\Delta\alpha \to \Delta\widehat{\beta}$ meet $\widehat{\beta} \odot \gamma = 1$. Such an example can be easily built for a practical block cipher, DES say. The differential $\Delta\alpha \to \Delta\beta$ contributes $\frac{1}{2}[(\frac{1}{2}+\epsilon)\times(\frac{1}{2}+\epsilon)+(\frac{1}{2}-\epsilon)\times(\frac{1}{2}-\epsilon)] = \frac{1}{4}+\epsilon^2$ to the probability of the distinguisher, and the other differentials $\Delta\alpha \to \Delta\widehat{\beta}$ contribute $\frac{1}{2}[(\frac{1}{2}+\epsilon)\times(\frac{1}{2}-\epsilon)+(\frac{1}{2}-\epsilon)\times(\frac{1}{2}+\epsilon)] = \frac{1}{4}-\epsilon^2$, which also cause a bias, but in a negative way, canceling the bias due to $\Delta\alpha \to \Delta\beta$. So the real bias of the distinguisher is 0, that is, the distinguisher has no cryptanalytic significance. However, by Biham et al.'s methodology, the bias of the distinguisher is $2 \times \frac{1}{2} \times \epsilon^2 = \epsilon^2$, and the distinguisher is useful (if ϵ^2 is large enough); but nevertheless in fact it is useless.

Therefore, Biham et al.'s methodology does not have the generality to describe differential-linear cryptanalysis.

3.4 New Methodology for Computing the Probability of a Differential-Linear Distinguisher

From the descriptions in Section 3.1 we know that a differential-linear distinguisher is concerned with the probability of equal output parities obtained by applying the bitwise dot product between mask δ and the ciphertexts of a pair of plaintexts with difference α; and its probability is dependent on mask γ (as well as the output difference β, but below we will see something different). We denote such a differential-linear distinguisher by $\langle \Delta\alpha, \Gamma\gamma\rangle \to \Gamma\delta$.

From a mathematical point, we make an analysis for the probability of a distinguisher (under the general assumptions). Out result is given as Theorem 1, followed by a proof.

Theorem 1. *An n-bit block cipher $\mathbb{E}$ is represented as a cascade of two sub-ciphers $\mathbb{E}_0$ and $\mathbb{E}_1$, where $\mathbb{E} = \mathbb{E}_0 \circ \mathbb{E}_1$. If $\Gamma\gamma \to \Gamma\delta$ is a linear approximation with bias ϵ for $\mathbb{E}_1$, α $(\neq 0)$ is an input difference for $\mathbb{E}_0$, and the probabilities for the differentials $\{\Delta\alpha \to \Delta\beta | \Pr_{\mathbb{E}_0}(\Delta\alpha \to \Delta\beta) > 0, \gamma \odot \beta = 0, \beta \in \{0,1\}^n\}$ is $\widehat{p}$ $(= \sum_{\gamma\odot\beta=0} \Pr_{\mathbb{E}_0}(\Delta\alpha \to \Delta\beta))$, then the probability of the differential-linear distinguisher $\langle\Delta\alpha, \Gamma\gamma\rangle \to \Gamma\delta$ is*

$$\Pr_{P\in\{0,1\}^n}(\mathbb{E}(P)\odot\delta = \mathbb{E}(P\oplus\alpha)\odot\delta) = \frac{1}{2} + 2(2\widehat{p}-1)\epsilon^2.$$

Proof. Given the input difference α for $\mathbb{E}_0$, there are one or more possible output differences $\{\beta | \Pr_{\mathbb{E}_0}(\Delta\ \alpha \to \Delta\beta) > 0, \beta \in \{0,1\}^n\}$; these output differences can be classified to two sets: one is $\{\beta|\gamma\odot\beta = 0, \Pr_{\mathbb{E}_0}(\Delta\alpha \to \Delta\beta) > 0, \beta\in\{0,1\}^n\}$, and the other is $\{\beta|\gamma\odot\beta = 1, \Pr_{\mathbb{E}_0}(\Delta\alpha\to\Delta\beta) > 0, \beta\in\{0,1\}^n\}$. If P is a plaintext chosen uniformly at random from $\{0,1\}^n$, then for a given difference from the set satisfying $\gamma\odot\beta = 0$, the probability $\Pr(\mathbb{E}(P)\odot\delta = \mathbb{E}(P\oplus\alpha)\odot\delta) = (\frac{1}{2}+\epsilon)\times(\frac{1}{2}+\epsilon)+(\frac{1}{2}-\epsilon)\times(\frac{1}{2}-\epsilon) = \frac{1}{2}+2\epsilon^2$; and for a given difference from the other set satisfying $\gamma\odot\beta = 1$, the probability $\Pr(\mathbb{E}(P)\odot\delta = \mathbb{E}(P\oplus\alpha)\odot\delta) = (\frac{1}{2}+\epsilon)\times(\frac{1}{2}-\epsilon)+(\frac{1}{2}-\epsilon)\times(\frac{1}{2}+\epsilon) = \frac{1}{2}-2\epsilon^2$. Finally, summing all the possibilities for β and $\gamma\odot\beta$ will result in the probability of the distinguisher. Therefore, the probability of a differential-linear distinguisher can be calculated in the following way.

$$\begin{aligned}
&\Pr(\mathbb{E}(P)\odot\delta = \mathbb{E}(P\oplus\alpha)\odot\delta)\\
=&\sum_{\beta\in\{0,1\}^n, Y\in\{0,1\}} \Pr(\mathbb{E}(P)\odot\delta = \mathbb{E}(P\oplus\alpha)\odot\delta, \mathbb{E}_0(P)\odot\gamma\oplus\mathbb{E}_0(P\oplus\alpha)\odot\gamma = Y, \mathbb{E}_0(P)\oplus\mathbb{E}_0(P\oplus\alpha) = \beta)\\
=&\sum_{\beta\in\{0,1\}^n, Y\in\{0,1\}} \Pr(\mathbb{E}(P)\odot\delta = \mathbb{E}(P\oplus\alpha)\odot\delta | \mathbb{E}_0(P)\odot\gamma\oplus\mathbb{E}_0(P\oplus\alpha)\odot\gamma = Y, \mathbb{E}_0(P)\oplus\mathbb{E}_0(P\oplus\alpha) = \beta)\times\\
&\qquad \Pr(\mathbb{E}_0(P)\odot\gamma\oplus\mathbb{E}_0(P\oplus\alpha)\odot\gamma = Y | \mathbb{E}_0(P)\oplus\mathbb{E}_0(P\oplus\alpha) = \beta)\times\Pr(\mathbb{E}_0(P)\oplus\mathbb{E}_0(P\oplus\alpha) = \beta)\\
=&\sum_{\beta\in\{0,1\}^n} \Pr(\mathbb{E}(P)\odot\delta = \mathbb{E}(P\oplus\alpha)\odot\delta | \mathbb{E}_0(P)\odot\gamma\oplus\mathbb{E}_0(P\oplus\alpha)\odot\gamma = 0, \mathbb{E}_0(P)\oplus\mathbb{E}_0(P\oplus\alpha) = \beta)\times
\end{aligned}$$

$$\Pr(\mathbb{E}_0(P)\odot\gamma\oplus\mathbb{E}_0(P\oplus\alpha)\odot\gamma=0|\mathbb{E}_0(P)\oplus\mathbb{E}_0(P\oplus\alpha)=\beta)\times\Pr(\mathbb{E}_0(P)\oplus\mathbb{E}_0(P\oplus\alpha)=\beta)+$$
$$\sum_{\beta\in\{0,1\}^n}\Pr(\mathbb{E}(P)\odot\delta=\mathbb{E}(P\oplus\alpha)\odot\delta|\mathbb{E}_0(P)\odot\gamma\oplus\mathbb{E}_0(P\oplus\alpha)\odot\gamma=1,\mathbb{E}_0(P)\oplus\mathbb{E}_0(P\oplus\alpha)=\beta)\times$$
$$\Pr(\mathbb{E}_0(P)\odot\gamma\oplus\mathbb{E}_0(P\oplus\alpha)\odot\gamma=1|\mathbb{E}_0(P)\oplus\mathbb{E}_0(P\oplus\alpha)=\beta)\times\Pr(\mathbb{E}_0(P)\oplus\mathbb{E}_0(P\oplus\alpha)=\beta)$$
$$=(\frac{1}{2}+2\epsilon^2)\times\sum_{\beta\in\{0,1\}^n,\gamma\odot\beta=0}\Pr(\mathbb{E}_0(P)\oplus\mathbb{E}_0(P\oplus\alpha)=\beta)+(\frac{1}{2}-2\epsilon^2)\times\sum_{\beta\in\{0,1\}^n,\gamma\odot\beta=1}\Pr(\mathbb{E}_0(P)\oplus\mathbb{E}_0(P\oplus\alpha)=\beta)$$
$$=\frac{1}{2}+2\epsilon^2\times(\sum_{\beta\in\{0,1\}^n,\gamma\odot\beta=0}\Pr(\mathbb{E}_0(P)\oplus\mathbb{E}_0(P\oplus\alpha)=\beta)-\sum_{\beta\in\{0,1\}^n,\gamma\odot\beta=1}\Pr(\mathbb{E}_0(P)\oplus\mathbb{E}_0(P\oplus\alpha)=\beta))$$
$$=\frac{1}{2}+2\epsilon^2\times(2\times\sum_{\beta\in\{0,1\}^n,\gamma\odot\beta=0}\Pr(\mathbb{E}_0(P)\oplus\mathbb{E}_0(P\oplus\alpha)=\beta)-1)$$
$$=\frac{1}{2}+2(2\widehat{p}-1)\epsilon^2.\ \square$$

Therefore, the bias of the differential-linear distinguisher $\langle\varDelta\alpha,\varGamma\gamma\rangle\rightarrow\varGamma\delta$ is $|\Pr_{P\in\{0,1\}^n}(\mathbb{E}(P)\odot\delta=\mathbb{E}(P\oplus\alpha)\odot\delta)-\frac{1}{2}|=2|2\widehat{p}-1|\epsilon^2$.

3.5 Implications

We should be cautious about using the assumption on a random distribution. Biham et al.'s methodology holds only when the assumption holds, and under the situation we have $\widehat{p}=p+(1-p)\frac{1}{2}=\frac{1}{2}+\frac{p}{2}$, meaning that the results obtained using Biham et al.'s and our methodologies are identical. From Theorem 1 we know that the distinguisher has the biggest bias $2\epsilon^2$ when $\widehat{p}=0$ or 1; and such an example is Langford and Hellman's 6-round differential-linear distinguisher of DES. When $\widehat{p}=p=\frac{1}{2}$, like the counterexample described in Section 3.3, then a significant distinction happens between Biham et al.'s methodology and ours: By Biham et al.'s methodology, the distinguisher is useful (if ϵ^2 is large enough), however, actually it is not useful.

Our result shows that using only one (truncated) differential satisfying $\beta\odot\gamma=0$ is not sufficient in some situations, and we should use all the differentials satisfying $\beta\odot\gamma=0$. This makes the distinguisher harder to construct, and may be infeasible in practice, due to a large number of possible output differences. Anyway, we should use at least those with a significant contribution to reduce the deviation if we are able to do so. In Biham et al.'s methodology, if the bias of the linear approximation keeps constant, the larger p is, the bigger is the bias of the distinguisher. Now, we know that may be not true: a differential with a bigger probability will not necessarily result in a distinguisher with a bigger bias.

When formatting a differential-linear distinguisher, in Biham et al.'s enhanced approach the attacker first chooses a (truncated) differential that meets the condition (as followed in [8,10,12,13], in practice the output difference of the differential has zeros in the bit positions concerned by the input mask of the linear approximation), then calculates the probability of the differential, and finally takes this probability as the value of p. The new methodology suggests a different format. Once the linear approximation and the input difference of the differentials are chosen, that how many rounds can be constructed for a distinguisher depends on the computational power available for the attacker, as demonstrated by our attacks on DES, CTC2 and Serpent described in the following three sections.

The notion of the related-key differential-linear analysis first appeared in [25], and it is a combination of related-key attacks [26,27] and differential-linear cryptanalysis. Building on Biham et al.'s enhanced differential-linear cryptanalysis, Kim [28] described an enhanced version of the related-key differential-linear analysis, and he got the formula for computing the probability of a related-key differential-linear distinguisher, similar to Biham et al.'s formula. As a result, we learn that this probability formula is not correct in some situations, and the general formula can be easily obtained as for a differential-linear distinguisher in Theorem 1.

4 Differential-Linear Cryptanalysis of Reduced DES

The DES block cipher is well known to both academia and industry. In 1994 Langford and Hellman [6] presented a 6-round differential-linear distinguisher of DES, and applied it to break 8-round DES. In 2002, using the enhanced technique Biham et al. [8] presented a 7-round differential-linear distinguisher of DES, and finally gave differential-linear attacks on 8 and 9-round DES; and an improved version of the 9-round attack appeared in pages 108–111 of [14].

In this section, we show that under the new methodology the 3-round linear approximation used in [6, 8] can be exploited to construct 7 and 8-round differential-linear distinguishers of DES; and the 8-round distinguisher can be used to break 10-round DES. More importantly, we are able to construct a 11-round differential-linear distinguisher of DES, and finally use it as the basis of a differential-linear attack on 13-round DES. See [3] for the specifications of DES. We write the subkey used in the S_l S-box of Round m as $K_{m,l}$, where $1 \leq m \leq 16, 1 \leq l \leq 8$.

4.1 A 7-Round Differential-Linear Distinguisher with Bias $2^{-7.94}$

Biham et al.'s 7-round differential-linear distinguisher [8, 14] consists of a 3-round linear approximation with bias 0.195 and a 4-round truncated differential with probability $\frac{12}{64}$. The 3-round linear approximation $\mathit{\Gamma}\gamma \rightarrow \mathit{\Gamma}\delta$ is $0x2104008000008000 \rightarrow 0x2104008000008000$, which was used for Langford and Hellman's differential-linear attack on 8-round DES. The 4-round (truncated) differential $\mathit{\Delta}\alpha \rightarrow \mathit{\Delta}\beta$ is $0x0000020240000000 \rightarrow 0x00W0XY0Z{*}{*}{*}{*}M{*}{*}{*}$[2], where $M \in \{0, 1, \cdots, 7\}$, $W, X \in \{0, 8\}$ and $Y, Z \in \{0, 2\}$. See Fig. 2 and Fig. 3 of [8] for an illustration of the differential and the linear approximation. $\mathit{\Delta}\beta$ has a zero difference at the five bit positions concerned by $\mathit{\Gamma}\gamma$, and thus meets $\beta \odot \gamma = 0$. Finally, they compute the bias of the distinguisher $\langle \mathit{\Delta}\alpha, \mathit{\Gamma}\gamma \rangle \rightarrow \mathit{\Gamma}\delta$ as $2 \times \frac{12}{64} \times 0.195^2 \approx 2^{-4.77}$. As discussed above, this is not sufficient, and we should consider all the possible differentials. The first round of the 4-round differential $\mathit{\Delta}\alpha \rightarrow \mathit{\Delta}\beta$ does not have a one probability, and thus there are a large number of possible output differences after 4-round DES, making it hard to count all the differentials.

To make things easier, we change the input difference $\mathit{\Delta}\alpha$ of the 4-round differential to $\mathit{\Delta}\widehat{\alpha} = 0x4000000000000000$ so that there is a one probability in the first round, thus reducing the number of possible output differences after 4-round DES. We still use the 3-round linear approximation $\mathit{\Gamma}\gamma \rightarrow \mathit{\Gamma}\delta$. Let's compute the probability of the new 7-round differential-linear distinguisher. Obviously, after the expansion $\mathbf{E}$ operation of the second round, $0x4$ in the input difference becomes $0x8$, which enters the S_1 S-box of the second round and generates 11 differences after the S-box: $\{\omega | \omega = 0x3, 0x5, 0x6, 0x7, 0x9, 0xA, 0xB, 0xC, 0xD, 0xE, 0xF\}$; the probabilities for these output differences are given in the second column of Table 2. We represent ω as a concatenation of four one-bit variables $a||b||c||d$, where $a, b, c, d \in \{0, 1\}$. Thus, the right half of the third round has the input difference $00000000a0000000b00000c0000000d0$ in binary notation, and this input difference can make at most 6 S-boxes of the third round active: $\mathrm{S}_2, \mathrm{S}_3, \mathrm{S}_4, \mathrm{S}_5, \mathrm{S}_6, \mathrm{S}_8$. After a simple analysis, we know that the left half of the input mask $\mathit{\Gamma}\gamma$ concerns the four bits of the output difference of the S_5 S-box of the third round, and we denote the four-bit output difference as $e||f||g||h$, where $e, f, g, h \in \{0, 1\}$. The right half of the input mask concerns the second most significant bit of the output difference of the S_1 S-box of the fourth round plus b. The input difference of the S_1 S-box of the fourth round depends on: (1) The second least significant bit of the output difference of the S_2 S-box of the third round, and we label the bit m; (2) The least significant bit of the output difference of the S_4 S-box of the third round, and we label the bit n; (3) The least significant bit (i.e., h) of the output difference of the S_5 S-box of the third round; (4) The most significant bit of the output difference of the S_6 S-box of the third round, and we label the bit p; (5) The most significant bit of the output difference of the S_8 S-box of the third round, and we label the bit q; and (6) The one-bit difference in $\mathit{\Delta}\widehat{\alpha}$. In summary, the five bits of the output difference concerned by the input mask $\mathit{\Gamma}\gamma$ depend on a total of 12 indeterminate one-bit differences: $a, b, c, d, e, f, g, h, m, n, p, q$. And the input difference of the S_1 S-box of the fourth round is $0||n||(m \oplus 1)||h||p||q$ in binary notation.

In the third round, the S_2 S-box has an input difference $00000a$ in binary notation, the S_4 S-box has an input difference $00000b$ in binary notation, the S_5 S-box has an input difference $0b0000$ in binary notation, the S_6 S-box has an input difference $000c00$ in binary notation, and the S_8 S-box has an input difference $000d00$ in binary notation. By the differential distribution tables of the S-boxes (see [22]), we compute the possible values as well as their probabilities for $m, n, p, q, (e||f||g||h)$, as follows: $\Pr_{\mathrm{S}_2}(m = 0|a = 0) = 1, \Pr_{\mathrm{S}_2}(m = 0|a = 1) = \frac{28}{64}, \Pr_{\mathrm{S}_2}(m = 1|a = 1) = \frac{36}{64}$, $\Pr_{\mathrm{S}_4}(n = 0|b = 0) = 1$, $\Pr_{\mathrm{S}_4}(n = 0|b = 1) = \frac{32}{64}, \Pr_{\mathrm{S}_4}(n = 1|b = 1) = \frac{32}{64}$, $\Pr_{\mathrm{S}_6}(p = 0|c = 0) = 1$, $\Pr_{\mathrm{S}_6}(p = 0|c = 1) = \frac{16}{64}, \Pr_{\mathrm{S}_6}(p = 1|c = 1) = \frac{48}{64}$, $\Pr_{\mathrm{S}_8}(q = 0|d = 0) = 1$, $\Pr_{\mathrm{S}_8}(q = 0|d = 1) = \frac{24}{64}, \Pr_{\mathrm{S}_8}(q = 1|d = 1) = \frac{40}{64}$, $\Pr_{\mathrm{S}_5}((e||f||g||h) = 0x0|b = 0) = 1$, $\Pr_{\mathrm{S}_5}((e||f||g||h) = 0x5|b = 1) = \frac{4}{64}, \Pr_{\mathrm{S}_5}((e||f||g||h) = 0x6|b = 1) = \frac{4}{64}$, $\Pr_{\mathrm{S}_5}((e||f||g||h) = 0x7|b = 1) = \frac{12}{64}$, $\Pr_{\mathrm{S}_5}((e||f||g||h) = 0x9|b = 1) = \frac{2}{64}$, $\Pr_{\mathrm{S}_5}((e||f||g||h) = 0xA|b = 1) = \frac{8}{64}$, $\Pr_{\mathrm{S}_5}((e||f||g||h) = 0xB|b = 1) = \frac{10}{64}$, $\Pr_{\mathrm{S}_5}((e||f||g||h) = 0xC|b = 1) = \frac{4}{64}$, $\Pr_{\mathrm{S}_5}((e||f||g||h) = 0xD|b = 1) = \frac{6}{64}$,

[2] This is after the exchange of the left and right halves in the fourth round.

Table 2. Probabilities for the eleven output differences in $\{\omega\}$

ω	$\mathrm{Pr}_{\mathrm{S}_1}(\varDelta 0x8 \to \varDelta\omega)$	$\mathrm{Pr}(\varDelta\beta_\omega \odot \varGamma\gamma = 0 \vert \varDelta 0x8 \to \varDelta\omega)$	$\mathrm{Pr}(\varDelta\widehat{\beta}_\omega \odot \varGamma\gamma = 0 \vert \varDelta 0x8 \to \varDelta\omega)$	$\mathrm{Pr}(\varDelta\widetilde{\beta}_\omega \odot \varGamma\widetilde{\gamma} = 0 \vert \varDelta 0x8 \to \varDelta\omega)$
$0x3$	$\frac{12}{64}$	0.55859375	0.50328584527596831	0.49779944866895676
$0x5$	$\frac{8}{64}$	0.50439453125	0.49747781828045845	0.49595199525356293
$0x6$	$\frac{8}{64}$	0.51708984375	0.50507303327322006	0.50433863041689619
$0x7$	$\frac{4}{64}$	0.50457763671875	0.49877615783771034	0.50256029706542904
$0x9$	$\frac{6}{64}$	0.578125	0.50051539158448577	0.50855094581311278
$0xA$	$\frac{2}{64}$	0.537109375	0.50116461620200425	0.50591027818154544
$0xB$	$\frac{8}{64}$	0.56123046875	0.49983475663202626	0.50239421910760029
$0xC$	$\frac{8}{64}$	0.4735107421875	0.49967876038863324	0.49929085310759547
$0xD$	$\frac{2}{64}$	0.4891510009765625	0.49995220528766993	0.49968796220765910
$0xE$	$\frac{2}{64}$	0.50665283203125	0.50015277066222552	0.50061782109781916
$0xF$	$\frac{4}{64}$	0.50272369384765625	0.50010005129477086	0.50005227406592345

$\mathrm{Pr}_{\mathrm{S}_5}((e||f||g||h) = 0xE|b = 1) = \frac{12}{64}$, $\mathrm{Pr}_{\mathrm{S}_5}((e||f||g||h) = 0xF|b = 1) = \frac{2}{64}$. We denote by t the second most significant bit of the output difference of the S_1 S-box of the fourth round, and by the differential distribution table of the S_1 S-box we compute the probability of $t = 0$ and 1 for all the input differences of the S-box; and the conditional probabilities are given in Table 3.

For each difference ω, we denote by β_ω the output difference(s) of the 4-round DES, and now we can compute the probability that the XOR of the concerned five bits of β_ω (whose values are $e, f, g, h, b \oplus t$) is zero (i.e., $\mathrm{Pr}(\varDelta\beta_\omega \odot \varGamma\gamma = 0|\varDelta 0x8 \to \varDelta\omega)$) by performing a computer program over all the possible (truncated) differential characteristics. These probabilities are given in the third column of Table 2. The largest number of possible differential characteristics happens when $\omega = 0xF$, which is $10 \times 2 \times 2 \times 2 \times 2 \times 2 \approx 2^{11.3}$, and a straightforward implementation takes a few seconds in a general PC.

Finally, by Theorem 1 we know that the probability of the 7-round distinguisher $\langle \varDelta\widehat{\alpha}, \varGamma\gamma \rangle \to \varGamma\delta$ is $\frac{1}{2} + 2 \times [2 \times \sum_\omega \mathrm{Pr}_{\mathrm{S}_1}(\varDelta 0x8 \to \varDelta\omega) \times \mathrm{Pr}(\varDelta\beta_\omega \odot \varGamma\gamma = 0|\varDelta 0x8 \to \varDelta\omega) - 1] \times 0.195^2 \approx \frac{1}{2} + 2 \times 2^{-4.22} \times 0.195^2 \approx \frac{1}{2} + 2^{-7.94}$. Thus, the distinguisher $\langle \varDelta\widehat{\alpha}, \varGamma\gamma \rangle \to \varGamma\delta$ has a bias of $2^{-7.94}$.

4.2 A 8-Round Differential-Linear Distinguisher with Bias $2^{-12.83}$

We obtain a 8-round differential-linear distinguisher of DES by appending one more round at the end of the four rounds covered by the differentials $\{\varDelta\widehat{\alpha} \to \beta_\omega\}$ in the above 7-round distinguisher. A detailed analysis reveals that the left half of the input mask $\varGamma\gamma$ concerns the four bits of the output difference of the S_5 S-box of the fourth round, and the right half of the input mask concerns the second most significant bit of the output difference of the S_1 S-box of the fifth round. Let $y_0, y_1, y_2, y_3, z \in \{0, 1\}$ be one-bit variables; we denote by $y_0||y_1||y_2||y_3$ the output difference of the S_5 S-box of the fourth round, and denote by z the second most significant bit of the output difference of the S_1 S-box of the fifth round. The input difference of the S_1 S-box of the fifth round depends on: (1) The second least significant bit of the output difference of the S_2 S-box of the fourth round, and we label the bit y_4; (2) The least significant bit of the output difference of the S_4 S-box of the fourth round, and we label the bit y_5; (3) The least significant bit (i.e., y_3) of the output difference of the S_5 S-box of the fourth round; (4) The most significant bit of the output difference of the S_6 S-box of the fourth round, and we label the bit y_6; (5) The most significant bit of the output difference of the S_7 S-box of the fourth round, and we label the bit y_7; and (6) The most significant bit of the output difference of the S_8 S-box of the fourth round, and we label the bit y_8.

The input difference of the S_2 S-box of the fourth round is $x_0||x_1||x_2||0||x_3||0$ (in binary notation), where x_0 denotes the most significant bit of the output difference of the S_6 S-box of the third round, x_1 denotes the most significant bit of the output difference of the S_8 S-box of the third round, x_2 denotes the least significant bit of the output difference of the S_3 S-box of the third round, and x_3 denotes the most significant bit of the output difference of the S_5 S-box of the third round. The input difference of the S_4 S-box of the fourth round is $0||x_4||x_5||x_6||x_7||0$, where x_4 denotes the most significant bit of the output difference of the S_2 S-box of the third round, x_5 denotes the second most significant bit of the output difference of the S_5 S-box of the third round, x_6 denotes the second least significant bit of the output difference of the S_8 S-box of the third round, and x_7 denotes the second most significant bit of the output difference of the S_3 S-box of the third round. The input difference of the S_5 S-box of the fourth

Table 3. Conditional Probabilities with $t = 0$ and 1

ξ	$\Pr_{\mathrm{S}_1}(\Delta t = 0 \vert \Delta(0 \Vert n \Vert (m \oplus 1) \Vert h \Vert p \Vert q) = \xi)$	$\Pr_{\mathrm{S}_1}(\Delta t = 1 \vert \Delta(0 \Vert n \Vert (m \oplus 1) \Vert h \Vert p \Vert q) = \xi)$
$0x0$	1	0
$0x1$	$\frac{32}{64}$	$\frac{32}{64}$
$0x2$	$\frac{28}{64}$	$\frac{36}{64}$
$0x3$	$\frac{36}{64}$	$\frac{28}{64}$
$0x4$	$\frac{20}{64}$	$\frac{44}{64}$
$0x5$	$\frac{28}{64}$	$\frac{36}{64}$
$0x6$	$\frac{28}{64}$	$\frac{36}{64}$
$0x7$	$\frac{36}{64}$	$\frac{28}{64}$
$0x8$	$\frac{28}{64}$	$\frac{36}{64}$
$0x9$	$\frac{28}{64}$	$\frac{36}{64}$
$0xA$	$\frac{32}{64}$	$\frac{32}{64}$
$0xB$	$\frac{32}{64}$	$\frac{32}{64}$
$0xC$	$\frac{24}{64}$	$\frac{40}{64}$
$0xD$	$\frac{40}{64}$	$\frac{24}{64}$
$0xE$	$\frac{36}{64}$	$\frac{28}{64}$
$0xF$	$\frac{24}{64}$	$\frac{40}{64}$
$0x10$	$\frac{24}{64}$	$\frac{40}{64}$
$0x11$	$\frac{36}{64}$	$\frac{28}{64}$
$0x12$	$\frac{32}{64}$	$\frac{32}{64}$
$0x13$	$\frac{32}{64}$	$\frac{32}{64}$
$0x14$	$\frac{32}{64}$	$\frac{32}{64}$
$0x15$	$\frac{32}{64}$	$\frac{32}{64}$
$0x16$	$\frac{40}{64}$	$\frac{24}{64}$
$0x17$	$\frac{32}{64}$	$\frac{32}{64}$
$0x18$	$\frac{32}{64}$	$\frac{32}{64}$
$0x19$	$\frac{32}{64}$	$\frac{32}{64}$
$0x1A$	$\frac{20}{64}$	$\frac{44}{64}$
$0x1B$	$\frac{28}{64}$	$\frac{36}{64}$
$0x1C$	$\frac{36}{64}$	$\frac{28}{64}$
$0x1D$	$\frac{28}{64}$	$\frac{36}{64}$
$0x1E$	$\frac{36}{64}$	$\frac{28}{64}$
$0x1F$	$\frac{36}{64}$	$\frac{28}{64}$

round is $x_7||0||x_8||x_9||x_{10}||x_{11}$, where x_8 denotes the least significant bit of the output difference of the S_2 S-box of the third round, x_9 denotes the least significant bit of the output difference of the S_6 S-box of the third round, x_{10} denotes the second most significant bit of the output difference of the S_4 S-box of the third round, and x_{11} denotes the least significant bit of the output difference of the S_8 S-box of the third round. The input difference of the S_6 S-box of the fourth round is $x_{10}||x_{11}||0||0||x_{12}||x_{13}$, where x_{12} denotes the most significant bit of the output difference of the S_3 S-box of the third round, and x_{13} denotes the second least significant bit of the output difference of the S_5 S-box of the third round. The input difference of the S_7 S-box of the fourth round is $x_{12}||x_{13}||x_{14}||x_{15}||x_{16}||x_{17}$, where x_{14} denotes the most significant bit of the output difference of the S_4 S-box of the third round, x_{15} denotes the second most significant bit of the output difference of the S_8 S-box of the third round, x_{16} denotes the second most significant bit of the output difference of the S_2 S-box of the third round, and x_{17} denotes the second most significant bit of the output difference of the S_6 S-box of the third round. The input difference of the S_8 S-box of the fourth round is $x_{16}||x_{17}||x_{18}||0||0||x_{19}$, where x_{18} denotes the second least significant bit of the output difference of the S_3 S-box of the third round, and x_{19} denotes the least significant bit of the output difference of the S_4 S-box of the third round.

The differential characteristics for the first two rounds are the same as in the 7-round distinguisher. In summary, the five bits of the output difference concerned by the input mask $\Gamma\gamma$ depend on a total of 33 indeterminate one-bit differences: $a, b, c, d, x_0, x_1, \cdots, x_{19}, y_0, y_1, \cdots, y_8$. For each difference ω, we denote by $\widehat{\beta_\omega}$ the output difference(s) of the 5-round DES. Now, similar to that described for the 7-round distinguisher we can compute the probability that the

XOR of the concerned five bits of $\widehat{\beta}_\omega$ (whose values are y_0, y_1, y_2, y_3, z) is zero (i.e., $\Pr(\varDelta\widehat{\beta}_\omega \odot \varGamma\gamma = 0|\varDelta 0x8 \rightarrow \varDelta\omega)$) by performing a computer program over all the possible (truncated) differential characteristics. These probabilities are given in the fourth column of Table 2. The largest number of possible differential characteristics happens also when $\omega = 0xF$, which is roughly $7 \times 10 \times 4 \times 6 \times 6 \times 10 \times 2^4 \times 2 \times 2 \times 2 \times 2 \times 2 \approx 2^{25.6}$; and it takes a few seconds to check in a PC.

Finally, by Theorem 1 the probability of the 8-round distinguisher $\langle \varDelta\widehat{\alpha}, \varGamma\gamma\rangle \rightarrow \varGamma\delta$ is $\frac{1}{2}+2\times[2\times\sum_\omega \Pr_{\mathrm{S}_1}(\varDelta 0x8 \rightarrow \varDelta\omega) \times \Pr(\varDelta\widehat{\beta}_\omega \odot \varGamma\gamma = 0|\varDelta 0x8 \rightarrow \varDelta\omega) - 1] \times 0.195^2 \approx \frac{1}{2} + 2 \times 2^{-9.11} \times 0.195^2 \approx \frac{1}{2} + 2^{-12.83}$. Therefore, the 8-round distinguisher $\langle \varDelta\widehat{\alpha}, \varGamma\gamma\rangle \rightarrow \varGamma\delta$ has a bias of $2^{-12.83}$.

The 8-round distinguisher $\langle \varDelta\widehat{\alpha}, \varGamma\gamma\rangle \rightarrow \varGamma\delta$ can be used to break 10-round DES. See Appendix B for the attack procedure and complexity.

Remarks. We have checked the possibility of extending the 8-round distinguisher $\langle \varDelta\widehat{\alpha}, \varGamma\gamma\rangle \rightarrow \varGamma\delta$ to a 9-round distinguisher by appending one more round at the end of the five rounds covered by the differentials $\{\varDelta\widehat{\alpha} \rightarrow \beta_\omega\}$. Now the five bits of the output difference concerned by the input mask $\varGamma\gamma$ depend on a total of 65 indeterminate one-bit differences, and there are roughly $2^{55.6}$ possible differential characteristics for $\omega = 0xF$. This is computationally infeasible for our PC. Anyway, if there was a cluster of a few hundred computers available, we were able to compute the probability of the 9-round distinguisher during several days.

4.3 A 11-Round Differential-Linear Distinguisher with Bias $2^{-24.05}$

The new methodology enables us to construct a 11-round differential-linear distinguisher of DES. The input difference for the 11-round distinguisher is the same as used for the 7 and 8-round distinguishers, that is, $\varDelta\widehat{\alpha} = 0x4000000000000000$; and we use for the distinguisher the following 6-round linear approximation $\varGamma\widetilde{\gamma} \rightarrow \varGamma\widetilde{\delta}$ with bias $1.95\times 2^{-9} \approx 2^{-8.04}$: $0x0000000001040080 \rightarrow 0x2104008000008000$, (This is the best 6-round linear approximation given in [5]).

The differential characteristics for the first two rounds and the input differences for the 6 (possible) active S-boxes of the third round are the same as in the 7/8-round distinguisher. In the third round, we denote respectively by x_0, x_1, x_2 the most significant bit, the second most significant bit and the second least significant bit of the output difference of the S_2 S-box, by $x_3||x_4||x_5||x_6$ the output difference of the S_3 S-box, by x_7, x_8, x_9 the second most significant bit, the second least significant bit and the least significant bit of the output difference of the S_4 S-box, by $x_{10}||x_{11}||x_{12}||x_{13}$ the output difference of the S_5 S-box, by x_{14}, x_{15}, x_{16} the most significant bit, the second most significant bit and the second least significant bit of the output difference of the S_6 S-box, and by x_{17}, x_{18}, x_{19} the most significant bit, the second least significant bit and the least significant bit of the output difference of the S_8 S-box.

In the fourth round, the S_1 S-box has the input difference $0||x_9||(x_2 \oplus 1)||x_{13}||x_{14}||x_{17}$, and we denote by y_0 the second most significant bit of its output difference; the S_2 S-box has the input difference $x_{14}||x_{17}||x_6||0||x_{10}||0$, and we denote by y_1 the least significant bit of its output difference; the S_3 S-box has the input difference $x_{10}||0||x_8||x_{16}||0||x_0$, and we denote by y_2 the second most significant bit of its output difference; the S_4 S-box has the input difference $0||x_0||x_{11}||x_{18}||x_4||0$, and we denote by y_3 the second most significant bit of its output difference; the S_6 S-box has the input difference $x_7||x_{19}||0||0||x_3||x_{12}$, and we denote by y_4 the least significant bit of its output difference; the S_8 S-box has the input difference $x_1||x_{15}||x_5||0||0||x_9$, and we denote by y_5 the least significant bit of its output difference. Thus we have that the input difference of the S_5 S-box of the fifth round is $y_2||(y_0 \oplus b)||y_1||y_4||y_3||y_5$.

A simple analysis reveals that the three bits concerned by the input mask $\varGamma\widetilde{\gamma}$ depend on: (1) x_{10}, x_{11} and x_{12}; and (2) The three most significant bits of the output difference of the S_5 S-box of the fifth round; and we denote the XOR of the three bits by z.

For each difference ω, we denote by $\widetilde{\beta}_\omega$ the output difference(s) of the 5-round DES. Now, we can similarly compute the probability that the XOR of the concerned three bits of $\widetilde{\beta}_\omega$ (i.e., $x_{10} \oplus x_{11} \oplus x_{12} \oplus z$) is zero by performing a computer program over all the possible (truncated) differential characteristics. These probabilities are given in the fifth column of Table 2. The largest number of possible differential characteristics happens also when $\omega = 0xF$, which is $7 \times 10 \times 4 \times 10 \times 6 \times 7 \times \times 2^6 \times 2 \approx 2^{23.9}$; and it takes a few seconds to check in a PC.

Finally, we have that the probability of the 11-round distinguisher $\langle \varDelta\widehat{\alpha}, \varGamma\widetilde{\gamma}\rangle \rightarrow \varGamma\widetilde{\delta}$ is $\frac{1}{2}+2\times[2\times\sum_\omega \Pr_{\mathrm{S}_1}(\varDelta 0x8 \rightarrow \varDelta\omega) \times \Pr(\varDelta\widetilde{\beta}_\omega \odot \varGamma\widetilde{\gamma} = 0|\varDelta 0x8 \rightarrow \varDelta\omega) - 1] \times (2^{-8.04})^2 \approx \frac{1}{2} + 2 \times 2^{-8.98} \times (2^{-8.04})^2 \approx \frac{1}{2} + 2^{-24.05}$. Therefore, the 11-round distinguisher $\langle \varDelta\widehat{\alpha}, \varGamma\widetilde{\gamma}\rangle \rightarrow \varGamma\widetilde{\delta}$ has a bias of $2^{-24.05}$.

4.4 Attacking 12-Round DES

We can use the 11-round distinguisher $\langle \Delta\widehat{\alpha}, \Gamma\widetilde{\gamma}\rangle \to \Gamma\widetilde{\widetilde{\delta}}$ to break 12 rounds of DES; the attack is basically the version of the 10-round attack in Appendix B when the first round is removed. With a success probability of about 99%, the attack requires $2^{50.6}$ pairs of chosen plaintexts with difference $\widehat{\alpha}$, and has a time complexity of $2 \times 2^{50.6} + 2 \times 2^{50.6} \times 2^6 \times \frac{1}{8\times 12} \approx 2^{52.34}$ 12-round DES encryptions.

4.5 Attacking 13-Round DES

The 11-round distinguisher $\langle \Delta\widehat{\alpha}, \Gamma\widetilde{\gamma}\rangle \to \Gamma\widetilde{\widetilde{\delta}}$ can be used to break 13-round DES. We assume the attacked rounds are the first thirteen rounds from Rounds 1 to 13. The attack procedure is as follows.

1. Choose $2^{47.27}$ structures S_i, $(i = 1, 2, \cdots, 2^{47.27})$, where a structure is defined to be a set of 2^4 plaintexts $P_{i,j}$ with bits (9,17,23, 31) of the left half taking all the possible values, bit (2) of the right half fixed to 0 and the other 59 bits fixed, $(j = 1, 2, \cdots, 2^4)$. In a chosen-plaintext attack scenario, obtain all the ciphertexts for the 2^4 plaintexts in each of the $2^{47.27}$ structures; we denote by $C_{i,j}$ the ciphertext for plaintext $P_{i,j}$.
2. Choose $2^{47.27}$ structures $\widehat{S}_i$, $(i = 1, \cdots, 2^{47.27})$, where a structure $\widehat{S}_i$ is obtained by setting 1 to bit (2) of the right half of all the plaintexts $P_{i,j}$ in S_i. In a chosen-plaintext attack scenario, obtain all the ciphertexts for the 2^4 plaintexts in each $\widehat{S}_i$.
3. Guess a value for $K_{1,1}$, and do as follows.
   (a) Initialize 2^{20} counters to zero, which correspond to the 2^{20} possible pairs of the 10 ciphertext bits: bit (17) of the left half and bits (1,2,3,4,5,8,14,25,32) of the right half.
   (b) Partially encrypt every (remaining) plaintext $P_{i,j}$ with the guessed $K_{1,1}$ to get its intermediate value immediately after Round 1; we denote it by $\varepsilon_{i,j}$.
   (c) Partially decrypt $\varepsilon_{i,j} \oplus 0x4000000000000000$ with the guessed $K_{1,1}$ to get its plaintext, and find the plaintext in $\widehat{S}_i$; we denote it by $\widehat{P}_{i,j}$, and denote by $\widehat{C}_{i,j}$ the corresponding ciphertext for $\widehat{P}_{i,j}$. Store $(C_{i,j}, \widehat{C}_{i,j})$ in a table.
   (d) For every ciphertext pair $(C_{i,j}, \widehat{C}_{i,j})$, increase 1 to the counter corresponding to the pair of the 10 ciphertext bits specified by $(C_{i,j}, \widehat{C}_{i,j})$.
   (e) Guess a value for $K_{13,1}$, and do as follows.
      i. For each of the 2^{20} pairs of the concerned 10 ciphertext bits, partially decrypt it with the guessed $K_{13,1}$ to get the pair of the 5 bits concerned by the output mask $\Gamma\widetilde{\widetilde{\delta}}$, and compute the XOR of the pair of the 5 bits (concerned by the output mask).
      ii. Count the number of the ciphertext pairs $(C_{i,j}, \widehat{C}_{i,j})$ such that the XOR of the pair of the 5 bits concerned by $\Gamma\widetilde{\widetilde{\delta}}$ is zero, and compute its deviation from $2^{50.27}$.
      iii. If the guess for $(K_{1,1}, K_{13,1})$ is the first guess for $(K_{1,1}, K_{13,1})$, then record the guess and the deviation computed in Step 2(e)(ii); otherwise, record the guess and its deviation only when the deviation is larger than that of the previously recorded guess, and remove the guess with the smaller deviation.
4. For the $K_{1,1}$ recorded in Step 2(e)(iii), exhaustively search for the remaining 48 key bits with two known plaintext/ciphertext pairs. If a 56-bit key is suggested, output it as the user key of the 13-round DES.

The attack requires $2^{52.27}$ chosen plaintexts. The required memory for the attack is dominated by the storage of the plaintexts and ciphertexts, which is $2^{52.27} \times 16 = 2^{56.27}$ bytes. Steps 1 and 2 have a time complexity of $2^{52.27}$ 13-round DES encryptions. Steps 3(b) and 3(c) have a time complexity of $2 \times 2^{51.27} \times 2^6 \times \frac{1}{8\times 13} \approx 2^{51.57}$ 13-round DES encryptions. Step 3(d) has a time complexity of $2^{51.27} \times 2^6 = 2^{57.27}$ memory accesses. The time complexity of Step 3(e) is dominated by the time complexity of Step 3(e)(i), which is $2 \times 2^6 \times 2^6 \times 2^{20} \times \frac{1}{8\times 13} \approx 2^{26.3}$ 13-round DES encryptions. Step 4 has a time complexity of 2^{48} 13-round DES encryptions. Therefore, the attack has a total time complexity of approximately $2^{52.97}$ 13-round DES encryptions and $2^{57.27}$ memory accesses. There are $2^{51.27}$ plaintext pairs $(P_{i,j}, \widehat{P}_{i,j})$ for a guess of $(K_{1,1}, K_{13,1})$, and thus following Theorem 2 of [29], we have that the attack has a success probability of about 99%.

5 Differential-Linear Cryptanalysis of CTC2

The CTC2 [15] cipher is designed to show the strength of the algebraic analysis [30] on block ciphers by the proposer, who described an algebraic attack on 6 rounds of the version of CTC2 that uses a 255-bit block size and a 255-bit key. In 2009, Dunkelman and Keller [13] described 6 and 7-round differential-linear distinguishers for the version of CTC2, and finally presented differential-linear attacks on 7 and 8 rounds of CTC2 (with a 255-bit block size and key). The attack on 8-round CTC2 is known as the best previously published result on CTC2 in terms of the numbers of attacked rounds.

In this section, using the new methodology we describe a 8.5-round differential-linear distinguisher with bias 2^{-68} for the CTC2 with a 255-bit block size and key, and give a differential-linear attack on 10-round CTC2 (with a 255-bit block size and a key). We first briefly review the CTC2 cipher.

5.1 The CTC2 Block Cipher

The CTC2 [15] block cipher has a variable block size, a variable length key, and a variable number of rounds. There are many combinations for the block size, key size and round number. As in [13], we only consider the version of CTC2 that uses a 255-bit block size and a 255-bit key. CTC2 uses the following two elementary operations to construct its round function.

- **S** is a non-linear substitution operation constructed by applying the same 3 × 3-bit bijective S-box 85 times in parallel to an input.
- **D** is a linear diffusion operation, which takes a 255-bit block $Y = (Y_{254}, \cdots, Y_1, Y_0)$ as input, and outputs a 255-bit block $Z = (Z_{254}, \cdots, Z_1, Z_0)$, computed as defined below.

$$\begin{cases} Z_{151} = Y_2 \oplus Y_{139} \oplus Y_{21} \\ Z_{(i \times 202+2) \bmod 255} = Y_i \oplus Y_{(i+137) \bmod 255} \qquad i = 0, 1, 3, 4, \cdots, 254 \end{cases}$$

CTC2 takes as input a 255-bit plaintext block P, and its encryption procedure for N_r rounds is, where $Z_0, X_i, Y_i, Z_i, X_{N_r}, Y_{N_r}, Z_{N_r}$ are 255-bit variables, and K_0, K_i, K_{N_r} are round keys generated from a user key K as $K_j = K \lll j$ in our notation, $(0 \le j \le N_r)$.

1. $Z_0 = P$.
2. For $i = 1$ to $N_r - 1$:
   - $X_i = Z_{i-1} \oplus K_{i-1}$,
   - $Y_i = \mathbf{S}(X_i)$,
   - $Z_i = \mathbf{D}(Y_i)$.
3. $X_{N_r} = Z_{N_r-1} \oplus K_{N_r-1}$, $Y_{N_r} = \mathbf{S}(X_i)$, $Z_{N_r} = \mathbf{D}(Y_{N_r})$.
4. Ciphertext $= Z_{N_r} \oplus K_{N_r}$.

The ith iteration of Step 2 in the above description is referred to below as Round i, $(1 \le i \le N_r - 1)$, and the transformations in Steps 3 and 4 are referred to below as Round N_r. This is in accordance with [15]. We number the 85 S-boxes in a round from 0 to 84 from right to left.

5.2 A 8.5-Round Differential-Linear Distinguisher with Bias 2^{-68}

The 8.5-round differential-linear distinguisher is made up of a 5.5-round linear approximation $\Gamma\gamma \rightarrow \Gamma\delta$ with bias 2^{-33} and all the 3-round differentials $\{\Delta\alpha \rightarrow \Delta\beta\}$ that meet $\beta \odot \gamma = 0$ with $\Delta\alpha = e_0$. The 5.5-round linear approximation $\Gamma\gamma \rightarrow \Gamma\delta$ is $e_{5,33,49,54,101,\ 112,131,138,155,168,188,193,217,247,251} \rightarrow e_{32,151}$, which is obtained by appending the following two rounds before the 3.5-round linear approximation $e_{14,104,134,241} \rightarrow e_{32,151}$ given in [13]: $e_{5,33,49,54,101,112,131,138,155,168,188,193,217,247,251} \xrightarrow{\mathbf{S}} e_{3,33,35,49,56,99,112,129,140,153,170,186,193,217,\ 247,249} \xrightarrow{\mathbf{D}} e_{38,53,\ 94,98,171,186,210,231} \xrightarrow{\mathbf{S}} e_{36,51,94,96,173,188,212,233} \xrightarrow{\mathbf{D}} e_{14,104,134,241}$. The input difference α is chosen so that there are only 16 active bit positions after being applied $\mathbf{D}^{-1}$. This enables us to conduct a differential-linear attack on 10-round CTC2 as presented in the next subsection. For any other one-bit difference except e_0, there are more than 50 active bit positions after applying $\mathbf{D}^{-1}$ to it, and thus the resulting distinguisher cannot be used to break 10-round CTC2, because too many subkey bits are required to guess.

Table 4. Probabilities for the four output differences in $\{\omega\}$

Difference (ω)	e_0	e_1	e_2	$e_{0,1,2}$
$\Pr_{\mathbf{S}}(\Delta\alpha \rightarrow \Delta\omega)$	2^{-2}	2^{-2}	2^{-2}	2^{-2}
$\Pr(\beta_\omega \odot \gamma = 0 \vert \Delta\alpha \rightarrow \Delta\omega)$	0.75	0.5	0.5	0.5

We now compute the probability of the 8.5-round differential-linear distinguisher. Without loss of generality, we assume that the 3-round differentials $\{\Delta\alpha \rightarrow \Delta\beta\}$ operate on Rounds 1 to 3, and the 5.5-round linear approximation $\Gamma\gamma \rightarrow \Gamma\delta$ operates on Rounds 4 to 9 (just before the **D** operation of Round 9). By the **D** operation, we learn that the input mask $\Gamma\gamma$ concerns the following 28 bit positions of the output difference of the **S** operation of Round 3: Bits 5, 12, 20, 24, 27, 45, 76, 82, 83, 86, 88, 89, 95, 105, 120, 142, 149, 157, 161, 163, 164, 200, 204, 206, 207, 220, 223 and 238.[3] The 28 concerned bit positions are covered in 24 S-boxes of Round 3: S-boxes 1, 4, 6, 8, 9, 15, 25, 27, 28, 29, 31, 35, 40, 47, 49, 52, 53, 54, 66, 68, 69, 73, 74 and 79; let Ω be the set of the 24 S-boxes.

On the other direction, the input difference $\Delta\alpha$ generates 4 possible differences after the **S** operation of Round 1: $\{\omega | \omega = e_0, e_1, e_2, e_{0,1,2}\}$, each with probability 2^{-2} as shown in the second row of Table 4. We represent the least significant three bits of ω as a concatenation of three one-bit variables $c||b||a$, where $a, b, c \in \{0, 1\}$. After the **D** operation of Round 1, a difference ω causes at most 6 active S-boxes of Round 2: S-boxes 0, 5, 23, 41, 50 and 68; the input difference for S-box 0 is $a||0||0$ in binary notation, the input difference for S-box 5 is $c||0||0$ in binary notation, the input difference for S-box 23 is $0||b||0$ in binary notation, the input difference for S-box 41 is $0||0||a$ in binary notation, the input difference for S-box 50 is $0||c||0$ in binary notation, and the input difference for S-box 68 is $0||0||b$ in binary notation. The 18-bit output difference of the 6 active S-boxes of Round 2 get involved in a total of 34 bits of the input difference of the S-box operation of Round 3: Bits 2, 7, 17, 21, 25, 36, 40, 49, 60, 64, 70, 78, 93, 102, 106, 113, 117, 121, 123, 128, 151, 155, 159, 170, 174, 181, 185, 204, 212, 223, 227, 234, 238 and 242. Among the 34 bits, only 8 bits are involved in the 24 S-boxes in Ω: Bits 25, 93, 106, 121, 159, 204, 223 and 238; and they are for S-boxes 8, 31, 35, 40, 53, 68, 74 and 79. The values for the 8 bits depend on 7 bits of the output difference of the 6 active S-boxes of Round 2: (1) The second least significant bit of the output difference of S-box 0, and we label the bit d; (2) The least and most significant bits of the output difference of S-box 5, and we denote them by e and f, respectively; (3) The second least significant bit of the output difference of S-box 23, and we label the bit g; (4) The most significant two bits of the output difference of S-box 50, and we denote them by h and m, respectively; and (5) The second least significant bit of the output difference of S-box 68, and we label the bit n.

As a result, we have: (I) S-box 8 of Round 3 has an input difference $0||h||0$, and the least significant bit, labeled by x_0, of its output difference is concerned by $\Gamma\gamma$; (II) S-box 31 of Round 3 has an input difference $0||0||e$, and the most significant bit, labeled by x_1, of its output difference is concerned by $\Gamma\gamma$; (III) S-box 35 of Round 3 has an input difference $0||m||0$, and the least significant bit, labeled by x_2, of its output difference is concerned by $\Gamma\gamma$; (IV) S-box 40 of Round 3 has an input difference $0||f||0$, and the least significant bit, labeled by x_3, of its output difference is concerned by $\Gamma\gamma$; (V) S-box 53 of Round 3 has an input difference $0||0||h$, and the most significant bit, labeled by x_4, of its output difference is concerned by $\Gamma\gamma$; (VI) S-box 68 of Round 3 has an input difference $0||0||d$, and the least and most significant bits, labeled respectively by x_5 and x_6, of its output difference are concerned by $\Gamma\gamma$; (VII) S-box 74 of Round 3 has an input difference $0||n||0$, and the second least significant bit, labeled by x_7, of its output difference is concerned by $\Gamma\gamma$; and (VIII) S-box 79 of Round 3 has an input difference $0||g||0$, and the second least significant bit, labeled by x_8, of its output difference is concerned by $\Gamma\gamma$. Now, whether $\beta \odot \gamma = 0$ is equivalent to whether $\bigoplus_{i=0}^{8} x_i = 0$.

By the differential distribution table of the S-box, we get the possible values for $d, (f||e), g, (m||h), n, x_0, x_1, x_2, x_3, x_4, x_5 \oplus x_6, x_7, x_8$ and the conditional probabilities, as follows: $\Pr(d = 0|a = 0) = 1, \Pr(d = 0|a = 1) = 0.5, \Pr(d = 1|a = 1) = 0.5, \Pr((f||e) = 0|c = 0) = 1, \Pr((f||e) = 1|c = 1) = 0.5, \Pr((f||e) = 3|c = 1) = 0.5, \Pr(g = 0|b = 0) = 1, \Pr(g = 1|b = 1) = 1, \Pr((m||h) = 0|c = 0) = 1, \Pr((m||h) = 1|c = 1) = 0.5, \Pr((m||h) = 3|c = 1) = 0.5, \Pr(n = 0|b = 0) = 1, \Pr(n = 0|b = 1) = 0.5, \Pr(n = 1|b = 1) = 0.5, \Pr(x_0 = 0|h = 0) = 1, \Pr(x_0 = 0|h = 1) = 0.5, \Pr(x_0 = 1|h = 1) = 0.5, \Pr(x_1 = 0|e = 0) = 1, \Pr(x_1 = 0|e = 1) = 0.5, \Pr(x_1 = 1|e = 1) = 0.5, \Pr(x_2 = 0|m = 0) = 1, \Pr(x_2 = 0|m = 1) = 0.5, \Pr(x_2 = 1|m = 1) = 0.5, \Pr(x_3 = 0|f = 0) = 1, \Pr(x_3 = 0|f = 1) = 0.5, \Pr(x_3 = 1|f = 1) = 0.5, \Pr(x_4 = 0|h = 0) = 1, \Pr(x_4 = 0|h = 1) = 0.5, \Pr(x_4 = 1|h = 1) = 0.5, \Pr(x_7 = 0|n = 0) = 1, \Pr(x_7 = 1|n = 1) =$

[3] Bit position 213 appears twice, and thus cancels out.

$1, \Pr(x_8 = 0|g = 0) = 1, \Pr(x_8 = 1|g = 1) = 1, \Pr(x_5 \oplus x_6 = 0|d = 0) = 1, \Pr(x_5 \oplus x_6 = 0|d = 1) = 0.5, \Pr(x_5 \oplus x_6 = 1|d = 1) = 0.5$.

For each difference ω, we denote by β_ω the output difference(s) immediately after Round 3, and using the above conditional probabilities we compute the probability of $\bigoplus_{i=0}^{8} x_i = 0$ by performing a program over all the possible (truncated) differential characteristics, which takes a few seconds in a general PC. These probabilities are given in the third row of Table 4.

Thus, by Theorem 1 we have that the probability of the 8.5-round distinguisher is $\frac{1}{2} + 2 \times [2 \times \sum_{\omega} \Pr_{\mathbf{S}}(\Delta 0x1 \rightarrow \Delta\omega) \times \Pr(\Delta\beta_\omega \odot \Gamma\gamma = 0|\Delta 0x1 \rightarrow \Delta\omega) - 1] \times (2^{-33})^2 = \frac{1}{2} + 2 \times 2^{-3} \times 2^{-66} = \frac{1}{2} + 2^{-68}$. Therefore, the 8.5-round differential-linear distinguisher has a bias of 2^{-68}.

5.3 Attacking 10-Round CTC2 with a 255-Bit Block Size and Key

The above 8.5-round distinguisher enables us to construct a differential-linear attack breaking 10-round CTC2 when used with a 255-bit block size and key.

We assume the attacked rounds are the first ten rounds from Rounds 1 to 10; and we use the distinguisher from Rounds 2 until before the $\mathbf{D}$ operation of Round 10. As mentioned earlier, we learn that the input difference α propagates to 16 bit positions after the inverse of the $\mathbf{D}$ operation of Round 1: Bits 17, 21, 40, 59, 78, 97, 116, 135, 139, 154, 158, 177, 196, 215, 234 and 253. The 16 active bits correspond to 16 S-boxes of Round 0: S-boxes 5, 7, 13, 19, 26, 32, 38, 45, 46, 51, 52, 59, 65, 71, 78 and 84; let Θ be the set of the 16 S-boxes, and K_Θ be the 48 bits of K_0 corresponding to the 16 S-boxes in Θ. Another observation is that we do not need to guess the subkey bits from K_{10}, because the output mask $\Gamma\delta$ of the 8.5-round distinguisher concerns the intermediate value immediately after the $\mathbf{S}$ operation of Round 10, and for a pair of ciphertexts $(C, \widehat{C})$ the value of $\delta \odot \mathbf{D}^{-1}(C) \oplus \delta \odot \mathbf{D}^{-1}(\widehat{C})$ equals to $\delta \odot \mathbf{D}^{-1}(C \oplus \widehat{C})$, which is independent of K_{10}. The attack procedure is as follows.

1. Choose 2^{94} structures S_i, $(i = 0, 1, \cdots, 2^{94} - 1)$, where a structure is defined to be a set of 2^{48} plaintexts $P_{i,j}$ with the 48 bits for the S-boxes in Θ taking all the possible values and the other 207 bits fixed, $(j = 0, 1, \cdots, 2^{48} - 1)$. In a chosen-plaintext attack scenario, obtain all the ciphertexts for the 2^{48} plaintexts in each of the 2^{94} structures; we denote by $C_{i,j}$ the ciphertext for plaintext $P_{i,j}$.
2. Guess a value for K_Θ, and do as follows.
   (a) Partially encrypt every (remaining) plaintext $P_{i,j}$ with the guessed K_Θ to get its intermediate value immediately after the $\mathbf{S}$ operation of Round 1; we denote it by $\varepsilon_{i,j}$.
   (b) Take bitwise complements to bits (17, 21, 40, 59, 78, 97, 116, 135, 139, 154, 158, 177, 196, 215, 234, 253) of $\varepsilon_{i,j}$, and keep invariant the other bits of $\varepsilon_{i,j}$; we denote the resulting value by $\widehat{\varepsilon}_{i,j}$.
   (c) Partially decrypt $\widehat{\varepsilon}_{i,j}$ with the guessed K_Θ to get its plaintext, and find the plaintext in S_i; we denote it by $\widehat{P}_{i,j}$, and denote by $\widehat{C}_{i,j}$ the corresponding ciphertext for $\widehat{P}_{i,j}$.
   (d) For every ciphertext pair $(C_{i,j}, \widehat{C}_{i,j})$, compute the XOR of bits 32 and 151 of $\mathbf{D}^{-1}(C_{i,j} \oplus \widehat{C}_{i,j})$.
   (e) Count the number of the ciphertext pairs $(C_{i,j}, \widehat{C}_{i,j})$ which have a zero XOR between bits 32 and 151 of $\mathbf{D}^{-1}(C_{i,j} \oplus \widehat{C}_{i,j})$, and compute its deviation from 2^{140}.
   (f) If the guess for K_Θ is the first guess, record it and its deviation computed in Step 2(e); otherwise, record it and its deviation only when the deviation is larger than the previously recorded deviation, and discard the previously recorded guess and its deviation.
3. For the recorded K_Θ in Step 2(f), exhaustively search for the remaining 207 key bits with a known plaintext/ciphertext pair. If a 255-bit key is suggested, output it as the user key of CTC2.

The attack requires 2^{142} chosen plaintexts. The time complexity of Step 2 is dominated by the time complexity of Steps 2(a), 2(c) and 2(d), which is approximately $2 \times 2^{141} \times 2^{48} \times \frac{16}{85 \times 10} + 2^{141} \times 2^{48} \times \frac{1}{10} \approx 2^{186.2}$ 10-round CTC2 encryptions. Step 3 has a time complexity of 2^{207} 10-round CTC2 encryptions. Therefore, the attack has a total time complexity of 2^{207} 10-round CTC2 encryptions to find the 255-bit key. There are 2^{141} plaintext pairs $(P_{i,j}, \widehat{P}_{i,j})$ for a guess of K_Θ. Following Theorem 2 of [29], we learn that the probability that the correct guess for K_Θ is recorded in Step 2(f) is about 99.9%. Thus, the attack has a success probability of about 99.9%.

6 Differential-Linear Cryptanalysis of Reduced Serpent

The Serpent [31] block cipher is one of the five AES finalists, second to the Rijndael [32] cipher. In 2003, Biham et al. [10] described a 8-round differential-linear distinguisher of Serpent, and presented a differential-linear attack on

10-round Serpent with a 128-bit key; and they presented a 9-round differential-linear distinguisher of Serpent, and finally gave a differential-linear attack on 11-round Serpent with a 192/256-bit key. In 2008 Dunkelman et al. [12] presented improved 8 and 9-round differential-linear distinguishers of Serpent, and finally used them as the basis for differential-linear attacks on 10-round Serpent with a 128-bit key, 11-round Serpent with a 192-bit key and 12-round Serpent with a 256-bit key. The attacks on 10-round Serpent with a 128-bit key, 11-round Serpent with a 192-bit key and 12-round Serpent with a 192-bit key are known as the best previously published cryptanalytic results on Serpent in terms of the numbers of attacked rounds.

In this section, we first present a 9-round linear approximation with bias 2^{-51}, which can be used to conduct linear attacks on 10-round Serpent with a 128-bit key and 11-round Serpent with a 192/256-bit key. Then, building on a 6-round linear approximation obtained by truncating the 9-round linear approximation, we construct a 9-round differential-linear distinguisher with bias $2^{-59.41}$ under the new methodology, and finally use it to conduct differential-linear attacks on 10-round Serpent with a 128-bit key, 11-round Serpent with a 192-bit key and 12-round Serpent with a 256-bit key.

6.1 The Serpent Block Cipher

The Serpent [31] block cipher has a 128-bit block size, a variable length key of up to 256 bits, and a total of 32 rounds; short keys is used by appending one "1" bit to the most significant bit end, followed by as many "0" bits as required. Serpent uses the following elementary operations:

- **IP**/**FP** is the initial/final permutation; see [31] for their specifications.
- $\mathbf{S}_i$ is a non-linear substitution operation constructed by applying the same 4×4-bit bijective $S_{i \bmod 8}$ S-box 32 times in parallel to an input, $(0 \leq i \leq 31)$. Refer to [31] for specifications of the S-boxes $S_0, S_1, \cdots, S_7$.
- **L** is a linear diffusion operation, which takes as input a 128-bit block of four 32-bit words $X = (X_3, X_2, X_1, X_0)$, and outputs a 128-bit block of four 32-bit words $Y = (Y_3, Y_2, Y_1, Y_0)$, computed as follows.
  - $X_0 = X_0 \lll 13$,
  - $X_2 = X_2 \lll 3$,
  - $X_1 = X_0 \oplus X_1 \oplus X_2$,
  - $X_3 = X_3 \oplus X_2 \oplus (X_0 \ll 3)$,
  - $X_1 = X_1 \lll 1$,
  - $X_3 = X_3 \lll 7$,
  - $X_0 = X_0 \oplus X_1 \oplus X_3$,
  - $X_2 = X_2 \oplus X_3 \oplus (X_1 \ll 3)$,
  - $X_0 = X_0 \lll 5$,
  - $X_2 = X_2 \lll 22$,
  - $Y = (X_3, X_2, X_1, X_0)$.

Serpent takes as input a 128-bit plaintext block P, and its encryption procedure is, where $\widehat{B}_0, \widehat{B}_1, \cdots, \widehat{B}_{32}$ are 128-bit variables, and $K_0, K_1, \cdots, K_{32}$ are round keys.

1. $\widehat{B}_0 = \mathbf{IP}(P)$.
2. For $i = 0$ to 30:
   - $\widehat{B}_{i+1} = \mathbf{L}(\mathbf{S}_i(\widehat{B}_i \oplus K_i))$.
3. $\widehat{B}_{32} = \mathbf{S}_{31}(\widehat{B}_{31} \oplus K_{31}) \oplus K_{32}$.
4. Ciphertext $= \mathbf{FP}(\widehat{B}_{32})$.

We refer to below the ith iteration of Step 2 in the above description as Round i, $(0 \leq i \leq 30)$, and the transformations in Steps 3 and 4 as Round 31. This is in accordance with [31]. We number the 32 S-boxes of a round from 0 to 31 from right to left. For simplicity, we describe a state S in a Serpent encryption as four 32-bit words (s_3, s_2, s_1, s_0), and write it as $(s_{3,31}||s_{2,31}||s_{1,31}||s_{0,31})||\cdots||(s_{3,1}||s_{2,31}||s_{1,1}||s_{0,1})||(s_{3,0}||s_{2,0}||s_{1,0}||s_{0,0})$, where $s_{j,l}$ is the lth bit of s_j, $(0 \leq j \leq 3, 0 \leq l \leq 31)$. We write $K_{i,m}$ for the 4-bit subkey of K_i that corresponds to S-box m of Round i, $(0 \leq m \leq 31)$. As the **IP** and **FP** operations are simply linear diffusion transformations, we omit them in our analysis. We denote by Serpent-128/192/256 the versions of Serpent that respectively use 128, 192 and 256 key bits.

Table 5. A 9-Round Linear Approximation with Bias 2^{-51}

Operation	input mask	Probability
$\mathbf{S}_3$	$0xD0600D000030300000300000D00DB70B$	$\frac{1}{2} \pm 2^{-12}$
$\mathbf{L}$	$0x40C00400001010000010000040048208$	1
$\mathbf{S}_4$	$0x00400000000000000000000000080006$	$\frac{1}{2} \pm 2^{-7}$
$\mathbf{L}$	$0x00400000000000000000000000040008$	1
$\mathbf{S}_5$	$0x00400000000000000000000000000002$	$\frac{1}{2} \pm 2^{-5}$
$\mathbf{L}$	$0x00400000000000000000000000000008$	1
$\mathbf{S}_6$	$0x00000000000000000000000080000000$	$\frac{1}{2} \pm 2^{-3}$
$\mathbf{L}$	$0x00000000000000000000000040000000$	1
$\mathbf{S}_7$	$0x000000A0000100000000000000000000$	$\frac{1}{2} \pm 2^{-5}$
$\mathbf{L}$	$0x00000010000100000000000000000000$	1
$\mathbf{S}_0$	$0x0000000000000000000010000B0000A0$	$\frac{1}{2} \pm 2^{-6}$
$\mathbf{L}$	$0x00000000000000000000100001000010$	1
$\mathbf{S}_1$	$0x0010000B0000B0000A00000000000000$	$\frac{1}{2} \pm 2^{-7}$
$\mathbf{L}$	$0x00100001000010000100000000000000$	1
$\mathbf{S}_2$	$0x0000A0000000000010000B0000B0000B$	$\frac{1}{2} \pm 2^{-6}$
$\mathbf{L}$	$0x00001000000000005000010000100001$	1
$\mathbf{S}_3$	$0x000B0000B000030000B0200E00000010$	$\frac{1}{2} \pm 2^{-8}$
$\mathbf{L}$	$0x00080000800001000080700200000060$	1
output mask	$0xAD1804BAC022040318D22A22230FC240$	/

6.2 9-Round Linear Approximations with Bias 2^{-51}

In [31], the Serpent designers gave upper bounds for the biases of 6 and 7-round linear approximations, which are 2^{-28} and 2^{-34}, respectively. However, in 2001 Biham et al. [17] described a few 6 and 7-round linear approximations with bias 2^{-25} and 2^{-32}, respectively, which are obviously beyond the bounds given by the Serpent designers; and they presented 9-round linear approximations with bias 2^{-52}, and used them to break 10-round Serpent-128 and 11-round Serpent-192/256. By the most recent theory [29] on the success probability of linear cryptanalysis, Biham et al.'s linear attacks on 10-round Serpent-128 and 11-round Serpent-192/256 will respectively require $2^{120.6}$ and $2^{122.9}$ known plaintexts to have a success probability of about 99%. The 9-round linear approximations are the best previously known linear approximations on Serpent with a cryptanalytic significance.

We observe that 9-round linear approximations with bias 2^{-51} can be obtained by changing the masks in the first few rounds of Biham et al.'s 9-round linear approximation, and one of them is detailed in Table 5. When we take the six rounds from the $\mathbf{S}_4$ operation until immediately before the $\mathbf{S}_2$ operation and then optimize the input and output masks, we can get 6-round linear approximations with bias 2^{-23}. 7-round linear approximations with bias 2^{-30} can be obtained by taking the seven rounds from the $\mathbf{S}_4$ operation until immediately before the second $\mathbf{S}_3$ operation and then optimizing the input mask. And, we can get 8-round linear approximations with bias 2^{-37} by taking the last eight rounds optimizing the input mask. They have a larger bias than Biham et al.'s linear approximations.

Building on the 9-round linear approximation with bias 2^{-51}, we can conduct linear attacks on 10-round Serpent-128 and 11-round Serpent-192/256, similar to those described in [17]. To attack 10-round Serpent-128 we change the output mask for the second $\mathbf{S}_3$ operation to $0x00040000400001000040100100000010$, and thus the resulting 9-round linear approximation has the output mask $0x2010090800E0300C0A00002004010000$, with bias 2^{-57}; and there are 11 active S-boxes in the round following the round with the output mask. By Theorem 2 of [29], the attack on 10-round Serpent-128 requires $2^{118.6}$ known plaintexts to have a success probability of 99%, and has a time complexity of about $2^{44} \times 2^{45} \times \frac{11}{32 \times 10} \approx 2^{85}$ 10-round Serpent encryptions. Further, to attack 11-round Serpent-192/256 we then change the input mask for the 9-round distinguisher to $0xF0600F0000F0300000F00000F00F2202$, and thus the new 9-round linear approximation has a bias of 2^{-60}, and there are 24 active S-boxes in the round preceding the rounds with the input mask, (the output mask for the round is $0x1007905C2041030854725DE06C107AF4$). The attack requires $2^{126.9}$ known plaintexts to have a success probability of 99%, and has a time complexity of about $2^{96} \times 2^{96} \times \frac{24}{32 \times 11} + 2^{44} \times (2^{126.9} \times \frac{11}{32 \times 11} + 2^{96}) \approx 2^{189}$ 11-round Serpent encryptions.

Table 6. Probabilities for the six output differences in $\{\omega\}$

Difference (ω)	$\Pr_{S_2}(\varDelta 0xA \rightarrow \varDelta\omega)$	$\Pr(\varDelta\beta_\omega \odot \varGamma\gamma = 0 \vert \varDelta 0xA \rightarrow \varDelta\omega)$
$0x2$	2^{-3}	0.453125
$0x4$	2^{-3}	0.502197265625
$0x6$	2^{-2}	0.5
$0x8$	2^{-3}	0.49609375
$0xA$	2^{-2}	0.500732421875
$0xE$	2^{-3}	0.5

Nevertheless, the significance of the 9-round linear approximation does not only lie in these attacks, but more significantly, as to be described in the next subsection, using a 6-round linear approximation obtained from the 9-round linear approximation, we can construct a 9-round differential-linear distinguisher that enables us to nontrivially break 12-round Serpent-256, the best cryptanalytic result for Serpent.

6.3 A 9-Round Differential-Linear Distinguisher with Bias $2^{-59.41}$

We frist give the 9-round differential-linear distinguisher, and then introduce its construction strategies at the end of this subsection. The distinguisher is made up of a 6-round linear approximation $\varGamma\gamma \rightarrow \varGamma\delta$ with bias 2^{-27} for Rounds 5 to 10 and all the 3-round differentials $\{\varDelta\alpha \rightarrow \varDelta\beta\}$ for Rounds 2 to 4 that meet $\beta \odot \gamma = 0$ with $\varDelta\alpha = 0x000000A0000000000000000000000000$. The 5-round linear approximation $\varGamma\gamma \rightarrow \varGamma\delta$ is $0x00400000000000000000000000000002 \rightarrow 0x000B0000B000030000B0200E00000010$, which is from the $\mathbf{S}_5$ operation until immediately before the second $\mathbf{S}_3$ operation of the 9-round linear approximation given in Table 5.

Given the input difference $\varDelta\alpha$, there is only one active S-box among the 32 S_2 S-boxes of Round 2, which generates 6 possible output differences: $\{\omega \vert \omega = 0x2, 0x4, 0x6, 0x8, 0xA, 0xE\}$; the probabilities for these output differences are given in the second column of Table 6, and the differential distribution tables of the eight S-boxes are presented in [33]. We write ω as $d\vert\vert c\vert\vert b\vert\vert a$ in binary notation, where $a, b, c, d \in \{0, 1\}$. By the **LT** operation we know that the input mask $\varGamma\gamma$ concerns a total of 3 bits of the output differences of three S_4 S-boxes in Round 4: (i) The most significant bit of the output difference of S-box 0, and we label it x_0; (ii) The second most significant bit of the output difference of S-box 4, and we label it x_1; and (iii) The second most significant bit of the output difference of S-box 29, and we label it x_2.

The 6 possible output differences $\{\omega\}$ may affect at most 18 S-boxes of Round 3, and a simple analysis reveals that only fifteen of them relate to the input differences of the three S-boxes of Round 4 concerned by the input mask. We now focus on the fifteen S_3 S-boxes in Round 3. S-box 0 has an input difference $d000$ in binary notation, and we denote the most significant bit and the second most significant bit of its output difference by y_0, y_1, respectively. S-box 2 has an input difference $000c$ in binary notation, and we denote the second least significant bit of its output difference by y_2. S-box 3 has an input difference $c000$ in binary notation, and we denote the most significant bit, the second most significant bit and the second least significant bit of its output difference by y_3, y_4, y_5, respectively. S-box 4 has an input difference $0a00$ in binary notation, and we denote the second most significant bit and the least significant bit of its output difference by y_6, y_7, respectively. S-box 6 has an input difference $0a00$ in binary notation, and we denote the second least significant bit and the least significant bit of its output difference by y_8, y_9, respectively. S-box 7 has an input difference $00a0$ in binary notation, and we denote the most significant bit and the second most significant bit of its output difference by y_{10}, y_{11}, respectively. S-box 8 has an input difference $000c$ in binary notation, and we denote the least significant bit of its output difference by y_{12}. S-box 11 has an input difference $000a$ in binary notation, and we denote the second most significant bit and the least significant bit of its output difference by y_{13}, y_{14}, respectively. S-box 18 has an input difference $0c00$ in binary notation, and we denote the least significant bit of its output difference by y_{15}. S-box 21 has an input difference $000a$ in binary notation, and we denote the second most significant bit and the least significant bit of its output difference by y_{16}, y_{17}, respectively. S-box 22 has an input difference $0d00$ in binary notation, and we denote the most significant bit, the second most significant bit and the least significant bit of its output difference by y_{18}, y_{19}, y_{20}, respectively. S-box 25 has an input difference $0c00$ in binary notation, and we denote the most significant bit, the second most significant bit and the least significant bit of its output difference by y_{21}, y_{22}, y_{23}, respectively. S-box 26 has an input difference $00b0$ in binary

Table 7. Relevant Probabilities for S_3

$\Pr((y_0\|\|y_1) = 0x0\|d = 1) = 0.125$	$\Pr((y_0\|\|y_1) = 0x1\|d = 1) = 0.125$
$\Pr((y_0\|\|y_1) = 0x2\|d = 1) = 0.125$	$\Pr((y_0\|\|y_1) = 0x3\|d = 1) = 0.625$
$\Pr(y_2 = 0x0\|c = 1) = 0.5$	$\Pr(y_2 = 0x1\|c = 1) = 0.5$
$\Pr((y_3\|\|y_4\|\|y_5) = 0x1\|c = 1) = 0.125$	$\Pr((y_3\|\|y_4\|\|y_5) = 0x3\|c = 1) = 0.125$
$\Pr((y_3\|\|y_4\|\|y_5) = 0x4\|c = 1) = 0.125$	$\Pr((y_3\|\|y_4\|\|y_5) = 0x6\|c = 1) = 0.375$
$\Pr((y_3\|\|y_4\|\|y_5) = 0x7\|c = 1) = 0.25$	$\Pr((y_{10}\|\|y_{11}) = 0x1\|a = 1) = 0.25$
$\Pr((y_{10}\|\|y_{11}) = 0x2\|a = 1) = 0.5$	$\Pr((y_{10}\|\|y_{11}) = 0x3\|a = 1) = 0.25$
$\Pr((y_6\|\|y_7) = 0x0\|a = 1) = 0.125$	$\Pr((y_6\|\|y_7) = 0x1\|a = 1) = 0.125$
$\Pr((y_6\|\|y_7) = 0x2\|a = 1) = 0.375$	$\Pr((y_6\|\|y_7) = 0x3\|a = 1) = 0.375$
$\Pr((y_8\|\|y_9) = 0x0\|a = 1) = 0.125$	$\Pr((y_8\|\|y_9) = 0x1\|a = 1) = 0.125$
$\Pr((y_8\|\|y_9) = 0x2\|a = 1) = 0.375$	$\Pr((y_8\|\|y_9) = 0x3\|a = 1) = 0.375$
$\Pr(y_{12} = 0x0\|c = 1) = 0.25$	$\Pr(y_{12} = 0x1\|c = 1) = 0.75$
$\Pr((y_{13}\|\|y_{14}) = 0x1\|a = 1) = 0.25$	$\Pr((y_{13}\|\|y_{14}) = 0x2\|a = 1) = 0.25$
$\Pr((y_{13}\|\|y_{14}) = 0x3\|a = 1) = 0.5$	$\Pr((y_{16}\|\|y_{17}) = 0x1\|a = 1) = 0.25$
$\Pr((y_{16}\|\|y_{17}) = 0x2\|a = 1) = 0.25$	$\Pr((y_{16}\|\|y_{17}) = 0x3\|a = 1) = 0.5$
$\Pr(y_{15} = 0x0\|c = 1) = 0.5$	$\Pr(y_{15} = 0x1\|c = 1) = 0.5$
$\Pr((y_{18}\|\|y_{19}\|\|y_{20}) = 0x2\|d = 1) = 0.25$	$\Pr((y_{18}\|\|y_{19}\|\|y_{20}) = 0x3\|d = 1) = 0.25$
$\Pr((y_{18}\|\|y_{19}\|\|y_{20}) = 0x4\|d = 1) = 0.125$	$\Pr((y_{18}\|\|y_{19}\|\|y_{20}) = 0x5\|d = 1) = 0.125$
$\Pr((y_{18}\|\|y_{19}\|\|y_{20}) = 0x6\|d = 1) = 0.125$	$\Pr((y_{18}\|\|y_{19}\|\|y_{20}) = 0x7\|d = 1) = 0.125$
$\Pr((y_{21}\|\|y_{22}\|\|y_{23}) = 0x2\|c = 1) = 0.25$	$\Pr((y_{21}\|\|y_{22}\|\|y_{23}) = 0x3\|c = 1) = 0.25$
$\Pr((y_{21}\|\|y_{22}\|\|y_{23}) = 0x4\|c = 1) = 0.125$	$\Pr((y_{21}\|\|y_{22}\|\|y_{23}) = 0x5\|c = 1) = 0.125$
$\Pr((y_{21}\|\|y_{22}\|\|y_{23}) = 0x6\|c = 1) = 0.125$	$\Pr((y_{21}\|\|y_{22}\|\|y_{23}) = 0x7\|c = 1) = 0.125$
$\Pr((y_{24}\|\|y_{25}) = 0x0\|b = 1) = 0.125$	$\Pr((y_{24}\|\|y_{25}) = 0x1\|b = 1) = 0.375$
$\Pr((y_{24}\|\|y_{25}) = 0x2\|b = 1) = 0.125$	$\Pr((y_{24}\|\|y_{25}) = 0x3\|b = 1) = 0.375$
$\Pr((y_{26}\|\|y_{27}) = 0x1\|c = 1) = 0.25$	$\Pr((y_{26}\|\|y_{27}) = 0x2\|c = 1) = 0.5$
$\Pr((y_{26}\|\|y_{27}) = 0x3\|c = 1) = 0.25$	$\Pr((y_{28}\|\|y_{29}) = 0x1\|b = 1) = 0.25$
$\Pr((y_{28}\|\|y_{29}) = 0x2\|b = 1) = 0.5$	$\Pr((y_{28}\|\|y_{29}) = 0x3\|b = 1) = 0.25$

notation, and we denote the second most significant bit and the second least significant bit of its output difference by y_{24}, y_{25}, respectively. S-box 29 has an input difference $00c0$ in binary notation, and we denote the most significant bit and the second most significant bit of its output difference by y_{26}, y_{27}, respectively. S-box 31 has an input difference $000b$ in binary notation, and we denote the second most significant bit and the second least significant bit of its output difference by y_{28}, y_{29}, respectively.

As a result, we get the input differences of the concerned three S_4 S-boxes of Round 4: (1) The input difference for S-box 0 is $(y_{19} \oplus y_{21})||(y_1 \oplus y_2 \oplus y_3 \oplus y_{11} \oplus y_{17} \oplus y_{28})||(y_{15} \oplus y_{29})||(y_7 \oplus y_{25})$; (2) The input difference for S-box 4 is $(y_{24} \oplus y_{26})||(y_4 \oplus y_6 \oplus y_8 \oplus y_{10} \oplus y_{13} \oplus y_{23})||(y_1 \oplus y_5 \oplus y_{20})||(y_{12} \oplus y_{15} \oplus y_{16})$; and (3) The input difference for S-box 29 is $(y_9 \oplus y_{18})||(y_0 \oplus y_6 \oplus y_{15} \oplus y_{27} \oplus y_{29})||y_{22}||y_{14}$.

By the differential distribution table of the S_3 S-box, we get all the possible values for $(y_0||y_1)$, y_2, $(y_3||y_4||y_5)$, $(y_6||y_7)$, $(y_8||y_9)$, $(y_{10}||y_{11})$, y_{12}, $(y_{13}||y_{14})$, y_{15}, $(y_{16}||y_{17})$, $(y_{18}||y_{19}||\ y_{20})$, $(y_{21}||y_{22}||y_{23})$, $(y_{24}||y_{25})$, $(y_{26}||y_{27})$ and $(y_{28}||y_{29})$ as well as their probabilities, which are given in Table 7. By the differential distribution table of S_4, we compute the conditional probability that for every possible input difference, the concerned bit of the output difference of each concerned S_4 S-box in Round 4 is 0 or 1; the probabilities are given in Table 8.

For each difference ω, we denote by β_ω the output difference(s) of the 3-round Serpent. Subsequently, we compute the probability that the XOR of the concerned 3 bits of β_ω is zero (i.e., $\Pr(\Delta\beta_\omega \odot \Gamma\gamma = 0|\Delta 0xA \to \Delta\omega)$) by performing a program over all the possible (truncated) differential characteristics. These probabilities are given in the third column of Table 6. A straightforward implementation takes several seconds in a general PC.

Therefore, the probability of the 9-round differential-linear distinguisher is $\frac{1}{2} + 2 \times [2 \times \sum_\omega \Pr_{S_2}(\Delta 0xA \to \Delta\omega) \times \Pr(\Delta\beta_\omega \odot \Gamma\gamma = 0|\Delta 0xA \to \Delta\omega) - 1] \times (2^{-27})^2 \approx \frac{1}{2} - 2 \times 2^{-6.41} \times 2^{-54} = \frac{1}{2} - 2^{-59.41}$. Thus, the 9-round differential-linear distinguisher has a bias of $2^{-59.41}$.

We use several strategies to find the above distinguisher. First, 9 rounds are the most that we can build for a differential-linear distinguisher in a general PC. As mentioned in Section 6.2, the best currently known 7-round linear approximation has bias 2^{-30}; thus if we aim to build an useful distinguisher operating on 10 (or more) rounds by using such a 7-round linear approximation, then the bias for the distinguisher is $2 \times |2\widehat{p} - 1| \times (2^{-30})^2 = |2\widehat{p} - 1| \cdot 2^{-59}$,

Table 8. Relevant Probabilities for S_4

Difference (ω)	$\Pr(x_0 = 0\|\omega)$	$\Pr(x_0 = 1\|\omega)$	$\Pr(x_1(x_2) = 0\|\omega)$	$\Pr(x_1(x_2) = 1\|\omega)$
$0x0$	1	0	1	0
$0x1$	0.25	0.75	0.25	0.75
$0x2$	0.5	0.5	0.25	0.75
$0x3$	0.5	0.5	0.75	0.25
$0x4$	0.5	0.5	0.5	0.5
$0x5$	0.25	0.75	0.5	0.5
$0x6$	0.5	0.5	0.5	0.5
$0x7$	0.5	0.5	0.5	0.5
$0x8$	0.25	0.75	0.5	0.5
$0x9$	0.75	0.25	0.5	0.5
$0xA$	0.5	0.5	0.5	0.5
$0xB$	0.5	0.5	0.5	0.5
$0xC$	0.25	0.75	0.25	0.75
$0xD$	0.75	0.25	0.5	0.5
$0xE$	0.5	0.5	0.75	0.25
$0xF$	0.5	0.5	0.25	0.75

so the margin for the value of $|2\widehat{p} - 1|$ is tough for using 3-round differentials. Alternatively we may choose to use 4-round differentials, instead of using a 7-round linear approximation, however, there are a large number of possible differential characteristics for which calculating the probability of $\beta \odot \gamma = 0$ is beyond the computational power of a general PC. Second, to use a 9-round distinguisher to attack 12-round Serpent, we should use such an input difference for the distinguisher that makes a small number of active S-boxes for the two rounds preceding the distinguisher, ideally less than 32, meaning that a small number of unknown key bits are required to guess; another way is to append two more rounds after the distinguisher, however this usually needs to guess more subkey bits. Third, some 3-round differentials should involve as few as possible active bits, and the input mask should concern as few as possible output bits of the **S** operation of the preceding round. After having checked the biases of a number of 9-round distinguishers, we find: Generally speaking, the more active or concerned bits are involved, the more smaller is the bias of the distinguisher, and the distinguisher is more likely to be ineffective. The above 9-round differential-linear distinguisher is the best we have found under the strategies, where the input mask concerns only three output bits of the preceding $\mathbf{S}_4$ operation, and either of the two values $0x2$ and $0x8$ of ω makes only two active S_3 S-boxes in the following round.

6.4 Differential-Linear Attack on 12-Round Serpent-256

We can use the 9-round differential-linear distinguisher as the basis for a differential-linear attack breaking 12-round Serpent-256. We attack Rounds 0 to 11, and use the distinguisher from Rounds 2 to 10. The input difference α becomes $0x000000A20400080000000000000000000$ after being applied the reverse of the **LT** operation of Round 1, and the 5 active bits correspond to S-boxes 18, 22, 24 and 25 of Round 1. It makes 27 active S-boxes of Round 0: S-boxes 0, 2, 3, 4, 5, 6, 7, 9, 11, 12, 13, 15, 16, $\cdots$, 29 and 31; let Θ be the set of the 27 S-boxes, and K_Θ be the 108 bits of K_0 corresponding to the 27 S-boxes in Θ. The 16 bits concerned by the output mask correspond to S-boxes 1, 8, 11, 13, 18, 23 and 28 of Round 11. The attack procedure is as follows, where the values of parameters λ and ϕ will be specified in the subsequent analysis.

1. Choose λ structures S_i, $(i = 0, 1, \cdots, \lambda-1)$, where a structure is defined to be a set of 2^{108} plaintexts $P_{i,j}$ with the 108 bits for the 27 S-boxes in Θ taking all the possible values and the other 20 bits fixed, $(j = 0, 1, \cdots, 2^{108} - 1)$. In a chosen-plaintext attack scenario, obtain all the ciphertexts for the 2^{108} plaintexts in each of the λ structures; we denote by $C_{i,j}$ the ciphertext for plaintext $P_{i,j}$.
2. Guess a value for $(K_\Theta, K_{1,18}, K_{1,22}, K_{1,24}, K_{1,25})$, and do as follows.
   (a) Initialize 2^{56} counters to zero, which correspond to the 2^{56} possible pairs of the 28 ciphertext bits corresponding to S-boxes 1, 8, 11, 13, 18, 23 and 28 of Round 11.

(b) Partially encrypt every (remaining) plaintext $P_{i,j}$ with the guessed $(K_{\Theta}, K_{1,18}, K_{1,22}, K_{1,24}, K_{1,25})$ to get its intermediate value immediately after the **S** operation of Round 1; we denote it by $\varepsilon_{i,j}$.

(c) Compute $\varepsilon_{i,j} \oplus 0x000000A2040008000000000000000000$, and we denote the resulting value by $\widehat{\varepsilon}_{i,j}$.

(d) Partially decrypt $\widehat{\varepsilon}_{i,j}$ with the guessed $(K_{\Theta}, K_{1,18}, K_{1,22}, K_{1,24}, K_{1,25})$ to get its plaintext, and find the plaintext in S_i; we denote it by $\widehat{P}_{i,j}$, and denote by $\widehat{C}_{i,j}$ the corresponding ciphertext for $\widehat{P}_{i,j}$.

(e) For every ciphertext pair $(C_{i,j}, \widehat{C}_{i,j})$, increase 1 to the counter corresponding to the pair of the 28 ciphertext bits specified by $(C_{i,j}, \widehat{C}_{i,j})$.

(f) Guess a value for $(K_{12,1}, K_{12,8}, K_{12,11}, K_{12,13}, K_{12,18}, K_{12,23}, K_{12,28})$, and do as follows.

   i. For each of the 2^{56} pairs of the concerned 28 ciphertext bits, partially decrypt it with the guessed $(K_{12,1}, K_{12,8}, \cdots, K_{12,28})$ to get the pair of the 16 bits concerned by the output mask, and compute the XOR of the pair of the 16 bits (concerned by the output mask).

   ii. Count the number of the ciphertext pairs $(C_{i,j}, \widehat{C}_{i,j})$ such that the XOR of the pair of the 16 bits concerned by the output mask is zero, and compute its deviation from $\lambda \cdot 2^{107}$.

   iii. If the guess for $(K_{\Theta}, K_{1,18}, K_{1,22}, K_{1,24}, K_{1,25}, K_{12,1}, K_{12,8}, \cdots, K_{12,28})$ belong to the first ϕ guesses for $(K_{\Theta}, K_{1,18}, K_{1,22}, K_{1,24}, K_{1,25}, K_{12,1}, K_{12,8}, \cdots, K_{12,28})$, then record the guess and the deviation computed in Step 2(f)(ii); otherwise, record the guess and its deviation only when the deviation is larger than the smallest deviation of the previously recorded ϕ guesses, and remove the guess with the smallest deviation from the ϕ guesses.

3. For every recorded $(K_{\Theta}, K_{1,18}, K_{1,22}, K_{1,24}, K_{1,25})$ in Step 2(f)(iii), exhaustively search for the remaining 132 key bits with two known plaintext/ciphertext pairs. If a 256-bit key is suggested, output it as the user key of the 12-round Serpent.

The attack requires $\lambda \times 2^{108}$ chosen plaintexts. The required memory for the attack is dominated by the storage of the plaintexts and ciphertexts, which is $\lambda \times 2^{108} \times 32 = \lambda \times 2^{113}$ bytes. The time complexity of Step 2 is dominated by the time complexity of Steps 2(b), 2(d) and 2(f)(i), which is $\lambda \times 2 \times 2^{107} \times 2^{124} \times \frac{27+4}{32 \times 12} + 2 \times 2^{124} \times 2^{28} \times 2^{56} \times \frac{7}{32 \times 12} \approx \lambda \times 2^{228.37}$ 12-round Serpent encryptions. Step 3 has a time complexity of at most $\phi \times 2^{132}$ 12-round Serpent encryptions. There are $\lambda \times 2^{107}$ plaintext pairs $(P_{i,j}, \widehat{P}_{i,j})$ for a guess of $(K_{\Theta}, K_{1,18}, K_{1,22}, K_{1,24}, K_{1,25}, K_{12,1}, K_{12,8}, \cdots, K_{12,28})$. Following Theorem 2 of [29], we have that the probability that the correct guess of $(K_{\Theta}, K_{1,18}, K_{1,22}, K_{1,24}, K_{1,25}, K_{12,1}, K_{12,8}, \cdots, K_{12,28})$ is recorded in Step 2(f)(iii) is about 96.6% when $\lambda = 2^{18.8}$ and $\phi = 1$, and is about 98.8% when $\lambda = 2^{16.5}$ and $\phi = 2^{104}$. Thus, when $\lambda = 2^{16.5}$ and $\phi = 2^{104}$, with a success probability of about 98.8% the attack requires $2^{125.5}$ chosen plaintexts, and has a total time complexity of approximately $2^{244.9}$ 12-round Serpent encryptions.

6.5 Differential-Linear Attack on 11-Round Serpent-192

The 9-round differential-linear distinguisher enables us to break 11-round Serpent-192; the attack is basically the version of the above 12-round Serpent-256 attack when the first round is removed. Let $\phi = 1$, then we get similarly that with a success probability of about 99.5% the attack requires $2^{107.2}$ structures of 2^{16} plaintexts with the 16 bits for S-boxes 18, 22, 24 and 25 (of Round 1) taking all the possible values and the other 112 bits fixed, and has a time complexity of $2^{123.2} + 2 \times 2^{122.2} \times 2^{16} \times \frac{4}{32 \times 11} + 2 \times 2^{16} \times 2^{28} \times 2^{56} \times \frac{7}{32 \times 11} \approx 2^{132.8}$ 11-round Serpent encryptions to find the correct value for $(K_{1,18}, K_{1,22}, K_{1,24}, K_{1,25}, K_{12,1}, K_{12,8}, K_{12,11}, K_{12,13}, K_{12,18}, K_{12,23}, K_{12,28})$. Finally, for the recorded $(K_{12,1}, K_{12,8}, K_{12,11}, K_{12,13}, K_{12,18}, K_{12,23}, K_{12,28})$, we get the 192-bit key by performing an exhaustive search on the remaining 164 key bits, which takes 2^{164} 11-round Serpent encryptions.

We can reduce the time complexity using a different 9-round differential-linear distinguisher, which is obtained by changing the output mask of the $\mathbf{S}_2$ operation to $0x0000100000\ 0000007000010000100001$. Consequently, the output mask $\Gamma\delta$ becomes $0x000B0000B001\ 030220B0200C00400010$, and it now concerns 11 S-boxes in the following round. Since the bias for the linear approximation remains invariant, the distinguisher has a bias $2^{-59.41}$. Similarly, the attack (with $\phi = 1$) requires $2^{109.5}$ structures (as described above) to have a success probability of about 99%, and has a time complexity of $2^{125.5} + 2 \times 2^{124.5} \times 2^{16} \times \frac{4}{32 \times 11} + 2 \times 2^{16} \times 2^{44} \times 2^{88} \times \frac{11}{32 \times 11} \approx 2^{144}$ 11-round encryptions to find the correct $(K_{1,18}, K_{1,22}, K_{1,24}, K_{1,25}, K_{12,1}, K_{12,5}, K_{12,8}, K_{12,11}, K_{12,13}, K_{12,15}, K_{12,16}, K_{12,18}, K_{12,20}, K_{12,23}, K_{12,28})$. Given the correct $(K_{12,1}, K_{12,5}, \cdots, K_{12,28})$, an exhaustive search for the remaining 148 key bits takes 2^{148} 11-round encryptions. Therefore, the attack has a total time complexity of approximately $2^{144} + 2^{148} \approx 2^{148.1}$ 11-round Serpent encryptions to find the 192-bit key.

We can also perform an attack procedure without using the counters; it is a time-memory tradeoff to the first 11-round Serpent-192 attack. For every guess of $(K_{1,18}, K_{1,22}, K_{1,24}, K_{1,25}, K_{12,1}, K_{12,8}, \cdots, K_{12,28})$, we first get the ciphertext pairs as in Steps 2(b)–(d), and then partially decrypt every ciphertext pair to get the XOR of the pair of the 16 bits concerned by the output mask, and finally perform as in Steps 2(f)(ii), 2(f)(iii) and 3 (with $\phi = 1$). The resulting attack has a total time complexity of approximately $2 \times 2^{122.2} \times 2^{44} \times \frac{7}{32\times 11} + 2^{164} \approx 2^{164.3}$ 11-round Serpent encryptions.

6.6 Differential-Linear Attack on 10-Round Serpent-128

The 9-round differential-linear distinguisher can be used to break 10-round Serpent-128; the attack is basically the version of the 12-round Serpent-256 attack when the first two rounds are removed. Let $\phi = 1$; then with a success probability of about 99.2% the attack requires $2^{122.4}$ plaintext pairs with difference α, and has a time complexity of $2^{123.4} + 2 \times 2^{28} \times 2^{56} \times \frac{7}{32\times 10} \approx 2^{123.4}$ 10-round Serpent encryptions to find the correct value for $(K_{1,18}, K_{1,22}, K_{1,24}, K_{1,25})$. Given the recorded $(K_{1,18}, K_{1,22}, K_{1,24}, K_{1,25})$, we can get the 128-bit key by performing an exhaustive search on the remaining 112 key bits, which takes 2^{112} 10-round Serpent encryptions.

We also find some 8-round and several different 9-round differential-linear distinguishers that can be used to break 10-round Serpent-128, like the two described in the latter part of the next subsection. Nevertheless, among them only one differential-linear distinguisher can be used to break 10-round Serpent-128 without requiring the additional memory for similar counters, and it is 9-round. The 9-round differential-linear distinguisher consists of a 6.5-round linear approximation $\Gamma\gamma \rightarrow \Gamma\delta$ with bias 2^{-27} and all the 2.5-round differentials $\{\Delta\alpha \rightarrow \Delta\beta\}$ that meet $\beta \odot \gamma = 0$ with $\Delta\alpha = 0x00000000000000900000000000000000$. The 6.5-round linear approximation $\Gamma\gamma \rightarrow \Gamma\delta$ is $0x000B0000B000030000B0200E00000010 \rightarrow 0x00400000000000000000000000040008$, which operates on 6.5 consecutive rounds of the 9-round linear approximation given in Table 5 in the decryption direction, from the $\mathbf{L}$ operation immediately before the $\mathbf{S}_3$ operation until the $\mathbf{L}$ operation immediately after the $\mathbf{S}_4$ operation. The 2.5-round differentials $\{\Delta\alpha \rightarrow \Delta\beta\}$ operate on 2.5 consecutive rounds in the decryption direction, from the $\mathbf{S}_5$ S-boxes of Round 13 until the $\mathbf{S}_3$ S-boxes of Round 11. The distinguisher has a bias of $2^{-59.01}$; see Appendix A for details of computing the bias. An important property of the distinguisher is that $\Gamma\delta$ concerns only three S_4 S-boxes of the preceding round. Likewise, with a success probability of 99%, the attack requires $2^{121.2}$ ciphertext pairs with difference α, and has a total time complexity of $2^{122.2} + 2 \times 2^{121.2} \times 2^{12} \times \frac{3}{32\times 10} + 2^{112} \approx 2^{127.5}$ 10-round Serpent encryptions to find the 128-bit key, where we use $\phi = 1$. We can also use the distinguisher to break 11-round Serpent-192 by appending one round after the round with $\Delta\alpha$, but anyway cannot break 12-round Serpent-256, because there is a large number of required unknown subkey bits in the extra round.

It is worthy to note that in both the above 10-round Serpent-128 attacks we can reduce the data and time complexity by using a reasonably greater ϕ, while keeping the same success probability.

6.7 Remarks

We have computed the biases of a number of 9-round differential-linear distinguishers, and a few of them can be used to attack 10-round Serpent-128 or 11-round Serpent-192, but almost all of them cannot be used to nontrivially break 12-round Serpent-256. For instance, we compute the bias of the 9-round differential-linear distinguisher constructed by replacing the input difference of the 9-round differential-linear distinguisher described in Section 6.3 with $0x00000000000000400000000000000000$ and replacing the input mask with $0x002000000000000000000000000000$ 02 (the intermediate masks keep invariant). This input mask is also used as the input mask of the 6-round linear approximation in Biham et al.'s 9-round differential-linear distinguisher. The reason for choosing the input difference is because it causes a minimum number of active bits for the following $\mathbf{S}_3$ operation. As a result, we get the probability of $\beta \odot \gamma = 0$ is approximately 0.49988768994808197. Besides, we also checked the 9-round differential-linear distinguisher when the input difference is $0x00000000000000040000000000000000$ (and the input mask is $0x00200000000000000000000000000002$), and the probability of $\beta \odot \gamma = 0$ is approximately 0.50000306963920593. Both the distinguishers have a bias of smaller than 2^{-64}, and are not useful.

Interestingly, we find such a 9-round differential-linear distinguisher that for every possible difference ω the probability of $\beta_\omega \odot \gamma = 0$ is surprisingly close (or equal) to $\frac{1}{2}$, which is obtained by replacing the input difference of the 9-round differential-linear distinguisher described in Section 6.3 with $0x00000000000000400000000000000000$ and replacing the input mask with $0x00E0000000000000000000000000000E$ (the intermediate masks keep invariant). The reason for choosing the input mask is that it will make a 6-round linear approximation with bias 2^{-25} (the best previously known 6-round linear approximation), thus improving a factor of 4 over the one used above. The input difference generates four possible output differences after the active S_2 S-box: $\{\omega|\omega = 0x6, 0xA, 0xB, 0xD\}$, each with probability 2^{-2}; the probability of $\beta_\omega \odot \gamma = 0$ for $\omega \in \{0x6, 0xA\}$ is 0.5, the probability of $\beta_\omega \odot \gamma = 0$ for $\omega = 0xB$ is 0.50000000018189894, and the probability of $\beta_\omega \odot \gamma = 0$ for $\omega = 0xD$ is 0.50000000000499334. Hence, the total probability of $\beta \odot \gamma = 0$ is approximately 0.50000000004672309, surprisingly close to $\frac{1}{2}$.

The second best 9-round differential-linear distinguisher which we have found might be potentially used to break 12-round Serpent-256 is the one described in Section 6.3 with the input difference being replaced by $0x00000000000000$ 000000005000000000 (keeping the other parts unchanged). The probability of $\beta \odot \gamma = 0$ is approximately 0.4990295171 7376709, and thus the distinguisher has a bias of approximately $2 \times 2^{-9} \times (2^{-27})^2 = 2^{-62}$, larger than 2^{-64}. There are 5 active S-boxes in Round 1, and 28 active S-boxes in Round 0. Changing the input difference to $0x0A000000000000000000000000000000$, we obtain the third best 9-round differential-linear distinguisher that we have found might be potentially used to break 12-round Serpent-256. The probability of $\beta \odot \gamma = 0$ is approximately 0.49903964996337891, and the distinguisher has a bias of approximately $2 \times 2^{-9.03} \times (2^{-27})^2 = 2^{-62.03}$. There are 4 active S-boxes in Round 1, and 26 active S-boxes in Round 0. If they were used to attack 12-round Serpent-256, the resulting attacks would require almost the entire codebook to have an acceptable success probability.

7 Methodology for the High-Order Differential-Linear Analysis

In this section, we review the high-order differential-linear analysis presented in [9], give the methodology for computing the probability of a high-order differential-linear distinguisher, followed by a few implications, and finally comment on the previous cryptanalytic results.

7.1 Review of the High-Order Differential-Linear Analysis

The high-order differential-linear analysis was proposed in 2005 by Biham, Dunkelman and Keller [9] as a combination of high-order differential cryptanalysis [7,34] and differential-linear cryptanalysis. High-order differential cryptanalysis focuses on whether a particular relation exists between the ciphertexts of a set of plaintexts with certain structure. e.g., whether the XOR of the ciphertexts is equal to 0.

The high-order differential-linear analysis involves an attacker building a high-order differential-linear distinguisher. Such a distinguisher treats $\mathbb{E} = \mathbb{E}_0 \circ \mathbb{E}_1$, a cascade of two sub-ciphers $\mathbb{E}_0$ and $\mathbb{E}_1$; it consists of a linear approximation $\varGamma\gamma \rightarrow \varGamma\delta$ with bias ϵ for $\mathbb{E}_1$ and a high-order differential for $\mathbb{E}_0$ that predicts $\bigoplus_{i=0}^{m-1} \mathbb{E}_0(P_i) = 0$ with probability p, where $P_0, P_1, \cdots, P_{m-1}$ are a set of plaintexts. The distinguisher focuses on the event $\bigoplus_{i=0}^{m-1} \mathbb{E}(P_i) \odot \delta = 0$.

Finally, by assuming that $\bigoplus_{i=0}^{m-1} \mathbb{E}_0(P_i) \odot \gamma = 0$ holds with probability $\frac{1}{2}$ (i.e., $\bigoplus_{i=0}^{m-1} \mathbb{E}_0(P_i) \odot \gamma$ has a uniform distribution) for the cases when the high-order differential does not predict $\bigoplus_{i=0}^{m-1} \mathbb{E}_0(P_i) = 0$, Biham et al. deduced that the high-order differential-linear distinguisher has a probability $\Pr(\bigoplus_{i=0}^{m-1} \mathbb{E}(P_i) \odot \delta = 0) = [p + (1-p) \times \frac{1}{2}] \times (\frac{1}{2} + 2^{m-1}\epsilon^m) + [1 - p - (1-p) \times \frac{1}{2}] \times [1 - (\frac{1}{2} + 2^{m-1}\epsilon^m)] = \frac{1}{2} + 2^{m-1}p\epsilon^m$, using the following lemma they obtained.

Lemma 1 (from [9]). *Let $\{X_0, X_1, \cdots, X_{m-1}\}$ be a set of inputs to $\mathbb{E}_1$. If $\varGamma\gamma \rightarrow \varGamma\delta$ is a linear approximation with bias ϵ for $\mathbb{E}_1$, then (under standard independence assumptions)* $\Pr(\bigoplus_{i=0}^{m-1} X_i \odot \gamma = \bigoplus_{i=0}^{m-1} \mathbb{E}_1(X_i) \odot \delta) = \frac{1}{2} + 2^{m-1}\epsilon^m$.

For a random function, the expected probability of a high-order differential-linear distinguisher is $\frac{1}{2}$. Thus, if the bias $|\Pr(\bigoplus_{i=0}^{m-1} \mathbb{E}(P_i) \odot \delta = 0) - \frac{1}{2}| = 2^{m-1}p\epsilon^m$ is sufficiently large, the distinguisher can be used to distinguish $\mathbb{E}$ from a random function.

7.2 New Methodology for Computing the Probability of a High-Order Differential-Linear Distinguisher

First observe that no matter to what value $\bigoplus_{i=0}^{m-1} X_i \odot \gamma$ is fixed, Lemma 1 holds (under an assumption about the independent behaviors of the linear approximations for $X_0, X_1, \cdots, X_{m-1}$).

Similarly we find that Biham et al.'s methodology is not correct in some situations. An intuitive counterexample is when the high-order differential predicts $\bigoplus_{i=0}^{m-1} \mathbb{E}_0(P_i) = 0$ with probability $\frac{1}{2}$ and predicts $\bigoplus_{i=0}^{m-1} \mathbb{E}_0(P_i) = 1$ with probability $\frac{1}{2}$, which is similar to the one discussed in Section 3.3.

We start the reasoning for the new methodology with the event $\bigoplus_{i=0}^{m-1} \mathbb{E}_0(P_i) \odot \gamma = 0$. Suppose the event $\bigoplus_{i=0}^{m-1} \mathbb{E}_0(P_i) \odot \gamma = 0$ happens with probability $\widehat{p}$. If the event $\bigoplus_{i=0}^{m-1} \mathbb{E}_0(P_i) \odot \gamma = 0$ happens, then by Lemma 1 we know that $\Pr(\bigoplus_{i=0}^{m-1}\mathbb{E}(P_i)\odot\delta=0)=\frac{1}{2}+2^{m-1}\epsilon^m$. If the event does not happen, i.e., $\bigoplus_{i=0}^{m-1}\mathbb{E}_0(P_i)\odot\gamma=1$, then we get that $\Pr(\bigoplus_{i=0}^{m-1}\mathbb{E}(P_i)\odot\delta=0)=1-(\frac{1}{2}+2^{m-1}\epsilon^m)$. Therefore, we have

$$\begin{aligned}
&\Pr(\bigoplus_{i=0}^{m-1}\mathbb{E}(P_i)\odot\delta=0)\\
=&\Pr(\bigoplus_{i=0}^{m-1}\mathbb{E}_0(P_i)\odot\gamma=0)\times\Pr(\bigoplus_{i=0}^{m-1}\mathbb{E}(P_i)\odot\delta=0|\bigoplus_{i=0}^{m-1}\mathbb{E}_0(P_i)\odot\gamma=0)+\\
&\Pr(\bigoplus_{i=0}^{m-1}\mathbb{E}_0(P_i)\odot\gamma=1)\times\Pr(\bigoplus_{i=0}^{m-1}\mathbb{E}(P_i)\odot\delta=0|\bigoplus_{i=0}^{m-1}\mathbb{E}_0(P_i)\odot\gamma=1)\\
=&\widehat{p}\times(\frac{1}{2}+2^{m-1}\epsilon^m)+(1-\widehat{p})\times[1-(\frac{1}{2}+2^{m-1}\epsilon^m)]\\
=&\frac{1}{2}+2^{m-1}(2\widehat{p}-1)\epsilon^m.
\end{aligned}$$

Now we can give the following result.

Theorem 2. *An n-bit block cipher $\mathbb{E}$ is represented as a cascade of two sub-ciphers $\mathbb{E}_0$ and $\mathbb{E}_1$, where $\mathbb{E}=\mathbb{E}_0\circ\mathbb{E}_1$. Let $P_0,P_1,\cdots,P_{m-1}$ be a set of plaintexts. If $\Gamma\gamma\rightarrow\Gamma\delta$ is a linear approximation with bias ϵ for $\mathbb{E}_1$, and the equation $\bigoplus_{i=0}^{m-1}\mathbb{E}_0(P_i)\odot\gamma=0$ holds with probability $\widehat{p}$, then the probability of the high-order differential-linear distinguisher is $\Pr(\bigoplus_{i=0}^{m-1}\mathbb{E}(P_i)\odot\delta=0)=\frac{1}{2}+2^{m-1}(2\widehat{p}-1)\epsilon^m$.*

Thus, the bias of the high-order differential-linear distinguisher is $|\Pr(\bigoplus_{i=0}^{m-1}\mathbb{E}(P_i)\odot\delta=0)-\frac{1}{2}|=2^{m-1}|2\widehat{p}-1|\epsilon^m$.

7.3 Implications

When formulating a high-order differential-linear distinguisher, we should compute the probability of $\bigoplus_{i=0}^{m-1}\mathbb{E}_0(P_i)\odot\gamma=0$, instead of the probability of $\bigoplus_{i=0}^{m-1}\mathbb{E}_0(P_i)=0$. Biham et al.'s methodology holds only when the assumption on a random distribution holds, and does not have the generality to describe the high-order differential-linear analysis. Other implications include those similar to what have been described in Section 3.5.

7.4 Comments on Previous Cryptanalytic Results

In [9], Biham et al. described a generic high-order differential-linear distinguisher for Feistel ciphers with a bijective round function, by combining a linear approximation with a high-order differential with probability 1, and they then applied it to break 8 rounds of the FEAL [35] cipher. They also spotted some weak keys for the IDEA [36] cipher after mounting a high-order differential-linear attack.

In 2007, Biham et al. [11] presented a high-order differential-linear attack on 6-round IDEA, which is the best currently published result on IDEA in a single key attack scenario.

With probability 1 the high-order differentials used in these attacks predict that it is zero for $\bigoplus_{i=0}^{m-1}\mathbb{E}_0(P_i)$ or only for the bits of $\bigoplus_{i=0}^{m-1}\mathbb{E}_0(P_i)$ that are concerned by the linear approximations used. Finally, Biham et al. computed the bias of a distinguisher as $2^{m-1}\times 1\times\epsilon^m=2^{m-1}\epsilon^m$ by their formula. However, following the new methodology we should view the formula extremely sceptically; but nevertheless by Theorem 2 the correct bias is also equal to $2^{m-1}\epsilon^m$. This is due to the use of an extreme high-order differential that makes $\bigoplus_{i=0}^{m-1}\mathbb{E}_0(P_i)\odot\gamma=0$ (or only the concerned bits of the sum) with a one probability: when the high-order differential meets $\bigoplus_{i=0}^{m-1}\mathbb{E}_0(P_i)=0$ with a one probability; and under this situation, $\bigoplus_{i=0}^{m-1}\mathbb{E}_0(P_i)\odot\gamma=0$ is certain to hold, making the bias obtained by Biham et al.'s formula happen to equal the bias obtained using the general formula. Therefore, the attack procedures remain usable, though a questionable formula is used for computing the probability of a distinguisher.

8 Methodology for the Differential-Bilinear Analysis

In this section, we first briefly review the differential-bilinear analysis proposed in [9], give the probability of a differential-bilinear distinguisher, followed by its implications, and finally correct certain previous cryptanalytic results.

8.1 Review of the Differential-Bilinear Analysis

In 2005, Biham, Dunkelman and Keller [9] introduced a combination of bilinear cryptanalysis [30] and differential cryptanalysis, known as the differential-bilinear analysis. Bilinear cryptanalysis is an extension to linear cryptanalysis, and works typically for Feistel ciphers in a principle similar to that of linear cryptanalysis; it is based on the use of a so-called bilinear approximation. We refer the reader to [30] for an introduction of bilinear cryptanalysis.

The differential-bilinear analysis uses a differential-bilinear distinguisher as the basis. The distinguisher treats a Feistel cipher $\mathbb{E}$ as a cascade of two sub-ciphers $\mathbb{E}_0$ and $\mathbb{E}_1$, where $\mathbb{E} = \mathbb{E}_0 \circ \mathbb{E}_1$, and it involves a (truncated) differential $\varDelta\alpha \rightarrow \varDelta\beta$ for $\mathbb{E}_0$ and a bilinear approximation with bias ϵ for $\mathbb{E}_1$. The bilinear approximation has the following general form,

$$\begin{aligned} &X_L \odot \alpha_0 \times X_R \odot \beta_0 \oplus X_L \odot \gamma_0 \oplus X_R \odot \delta_0 \oplus Y_L \odot \alpha_1 \times Y_R \odot \beta_1 \oplus Y_L \odot \gamma_1 \oplus Y_R \odot \delta_1 = \\ &X_L \odot \zeta_0 \times K \odot \zeta_1 \oplus X_R \odot \eta_0 \times K \odot \eta_1 \oplus Y_L \odot \theta_0 \times K \odot \theta_1 \oplus Y_R \odot \mu_0 \times K \odot \mu_1 \oplus K \odot \nu, \end{aligned} \tag{1}$$

where X_L, X_R are respectively the left and right halves of a value X from $\{0,1\}^n$, Y_L, Y_R are respectively the left and right halves of $\mathbb{E}_1(X)$, $\alpha_0, \beta_0, \gamma_0, \delta_0, \alpha_1, \beta_1, \gamma_1, \delta_1, \zeta_0, \zeta_1, \eta_0, \eta_1, \theta_0, \theta_1, \mu_0, \mu_1, \nu$ are some n-bit masks, and K is the subkey.

The differential $\varDelta\alpha \rightarrow \varDelta\beta$ meets the following equation with probability p,

$$T_L \odot \alpha_0 \times T_R \odot \beta_0 \oplus T_L \odot \gamma_0 \oplus T_R \odot \delta_0 = T_L^* \odot \alpha_0 \times T_R^* \odot \beta_0 \oplus T_L^* \odot \gamma_0 \oplus T_R^* \odot \delta_0, \tag{2}$$

where T_L, T_R are respectively the left and right halves of the encryption result of $\mathbb{E}_0(P)$ with P being a randomly chosen plaintext, and T_L^*, T_R^* are respectively the left and right halves of $\mathbb{E}_0(P \oplus \alpha)$.

The differential-bilinear distinguisher considers the event expressed by Eq. (3).

$$C_L \odot \alpha_1 \times C_R \odot \beta_1 \oplus C_L \odot \gamma_1 \oplus C_R \odot \delta_1 = C_L^* \odot \alpha_1 \times C_R^* \odot \beta_1 \oplus C_L^* \odot \gamma_1 \oplus C_R^* \odot \delta_1, \tag{3}$$

where C_L, C_R are respectively the left and right halves of $\mathbb{E}(P)$, and C_L^*, C_R^* are respectively the left and right halves of $\mathbb{E}(P \oplus \alpha)$.

As in differential-linear cryptanalysis, Biham et al. got that Eq. (3) holds with probability $(\frac{1}{2}+\epsilon)\times(\frac{1}{2}+\epsilon)+(\frac{1}{2}-\epsilon)\times(\frac{1}{2}-\epsilon) = \frac{1}{2}+2\epsilon^2$ when the differential meets the condition given as Eq. (2); and they assumed Eq. (2) holds with probability $\frac{1}{2}$ for the cases when the differential does not meet Eq. (2). Finally, they concluded that the differential-bilinear distinguisher has a probability $\Pr(\mathit{Eq.}\ (3)\ \mathit{holds}) = [p+(1-p)\frac{1}{2}]\times(\frac{1}{2}+2\epsilon^2)+(\frac{1}{2}-\frac{1}{2}p)\times(\frac{1}{2}-2\epsilon^2) = \frac{1}{2}+2p\epsilon^2$.

For a random function, the expected probability of a differential-bilinear distinguisher is $\frac{1}{2}$. Thus, if the bias $|\Pr(\mathit{Eq.}\ (3)\ \mathit{holds}) - \frac{1}{2}| = 2p\epsilon^2$ is sufficiently large, the distinguisher can be used to distinguish $\mathbb{E}$ from a random function.

Some noteworthy particulars for differential-bilinear cryptanalysis were discussed in [9,14], not existing in differential-linear cryptanalysis; for example, those bilinear terms in Eq. (1) multiplied by the dot product of the subkey and the masks and those bilinear terms in Eq. (2). It is obvious that only knowing the difference β between $(T_L||T_R)$ and $(T_L^*||T_R^*)$ is not sufficient to compute Eq. (2). If $(T_L||T_R) \oplus (T_L^*||T_R^*) = \beta = (\beta_L, \beta_R)$, then $T_L \odot \alpha_0 \times T_R \odot \beta_0 \oplus T_L^* \odot \alpha_0 \times T_R^* \odot \beta_0 = T_L \odot \alpha_0 \times \beta_R \odot \beta_0 \oplus \beta_L \odot \alpha_0 \times T_R \odot \beta_0 \oplus \beta_L \odot \alpha_0 \times \beta_R \odot \beta_0$. Thus when the chosen α_0, β_0, β are required to meet $\beta_L \odot \alpha_0 = \beta_R \odot \beta_0 = 0$, then Eq. (2) can be computed as an expression involving only $\gamma_0, \delta_0, \beta$, i.e., $\beta_L \odot \gamma_0 \oplus \beta_R \odot \delta_0 = 0$, without requiring the values of $(T_L||T_R)$ and $(T_L^*||T_R^*)$.

8.2 New Methodology for Computing the Probability of a Differential-Bilinear Distinguisher

Similarly it is easy to build an intuitive counterexample to Biham et al.'s methodology.

To simplify our following descriptions, we write a bilinear approximation expressed by Eq. (1) as $(\varGamma\alpha_0, \varGamma\beta_0, \varGamma\gamma_0, \varGamma\delta_0) \longrightarrow (\varGamma\alpha_1, \varGamma\beta_1, \varGamma\gamma_1, \varGamma\delta_1)$ if there is no ambiguity about the cipher in use; otherwise, we will give an explicit statement for the cipher. We denote by $\langle\varDelta\alpha, \varGamma\alpha_0, \varGamma\beta_0, \varGamma\gamma_0, \varGamma\delta_0\rangle \rightarrow (\varGamma\alpha_1, \varGamma\beta_1, \varGamma\gamma_1, \varGamma\delta_1)$ a differential-bilinear distinguisher that focuses on the event expressed by Eq. (3).

Similar to the reasonings given in the last two sections, if the event Eq. (2) happens with probability $\widehat{p}$, then the event Eq. (3) occurs with probability $(\frac{1}{2}+\epsilon)\times(\frac{1}{2}+\epsilon)+(\frac{1}{2}-\epsilon)\times(\frac{1}{2}-\epsilon) = \frac{1}{2}+2\epsilon^2$; and if the event Eq. (2) does not happen, then the event Eq. (3) occurs with probability $(\frac{1}{2}+\epsilon)\times(\frac{1}{2}-\epsilon)+(\frac{1}{2}-\epsilon)\times(\frac{1}{2}+\epsilon) = \frac{1}{2}-2\epsilon^2$. Therefore, the event Eq. (3) occurs with a total probability $\widehat{p}\times(\frac{1}{2}+2\epsilon^2)+(1-\widehat{p})\times(\frac{1}{2}-2\epsilon^2) = \frac{1}{2}+2(2\widehat{p}-1)\epsilon^2$. Thus we have the following result.

Theorem 3. *An n-bit block cipher $\mathbb{E}$ is represented as a cascade of two sub-ciphers $\mathbb{E}_0$ and $\mathbb{E}_1$, where $\mathbb{E} = \mathbb{E}_0 \circ \mathbb{E}_1$. If $(\varGamma\alpha_0, \varGamma\beta_0, \varGamma\gamma_0, \varGamma\delta_0) \rightarrow (\varGamma\alpha_1, \varGamma\beta_1, \varGamma\gamma_1, \varGamma\delta_1)$ is a bilinear approximation with bias ϵ for $\mathbb{E}_1$, α $(\neq 0)$ is an input difference for $\mathbb{E}_0$, and Eq. (2) holds with probability $\widehat{p}$, then the probability of the differential-bilinear distinguisher $\langle\varDelta\alpha, \varGamma\alpha_0, \varGamma\beta_0, \varGamma\gamma_0, \varGamma\delta_0\rangle \rightarrow (\varGamma\alpha_1, \varGamma\beta_1, \varGamma\gamma_1, \varGamma\delta_1)$ is $\Pr(Eq.\ (3)\ holds) = \frac{1}{2} + 2(2\widehat{p} - 1)\epsilon^2$.*

Hence, the bias of the differential-bilinear distinguisher $\langle\varDelta\alpha, \varGamma\alpha_0, \varGamma\beta_0, \varGamma\gamma_0, \varGamma\delta_0\rangle \rightarrow (\varGamma\alpha_1, \varGamma\beta_1, \varGamma\gamma_1, \varGamma\delta_1)$ is $|\Pr(Eq.\ (3)\ holds) - \frac{1}{2}| = 2|2\widehat{p} - 1|\epsilon^2$.

8.3 Implications

When formulating a differential-bilinear distinguisher, we should compute the probability that Eq. (2) holds for all the possible differentials, instead of the probability that only the differential $\varDelta\alpha \rightarrow \varDelta\beta$ meets Eq. (2). Biham et al.'s methodology holds only when the assumption on a random distribution holds, and does not have the generality to describe the differential-bilinear analysis. Other implications are similar to those described in Section 3.5.

8.4 Correcting Previous Differential-Bilinear Attack on 8-Round DES

In [9] Biham et al. applied the differential-bilinear analysis technique to break 8-round DES. The 8-round DES attack uses a 6-round differential-bilinear distinguisher that consists of a 3-round bilinear approximation with bias 1.66×2^{-3} and a 3-round truncated differential that with probability $\frac{23}{32}$ has a zero output difference for the bits concerned by the bilinear approximation (see Fig. 1 of [9] for its details). Taken from [30], the 3-round bilinear approximation for the distinguisher is $X_L \odot \alpha_0 \times X_R \odot \beta_0 \oplus X_L \odot \gamma_0 \oplus X_R \odot \delta_0 \oplus Y_L \odot \alpha_1 \times Y_R \odot \beta_1 \oplus Y_L \odot \gamma_1 \oplus Y_R \odot \delta_1 = K \odot \nu \oplus X_L \odot \alpha_0 \times K \odot \mu \oplus Y_L \odot \alpha_0 \times K \odot \tau$, where $\alpha_0 = \alpha_1 = 0x20000000, \beta_0 = \beta_1 = 0x00019000, \gamma_0 = \gamma_1 = 0x21040080, \delta_0 = \delta_1 = 0x00008000$. The 3-round differential $\varDelta\alpha \rightarrow \varDelta\beta$ is $(\varDelta 0x0020000000000000 \rightarrow \varDelta 0x***MR***0X00NZ0Y)$, where $X, Y \in \{0, 4\}, Z \in \{0, 1\}, N \in \{0, 8\}, M \in \{0, 2, 4, 6, 8, A, C, E\}, R \in \{2, 4, 6\}$. Finally, Biham et al. computed the bias of the distinguisher $\langle\varDelta\alpha, \varGamma\alpha_0, \varGamma\beta_0, \varGamma\gamma_0, \varGamma\delta_0\rangle \rightarrow (\varGamma\alpha_1, \varGamma\beta_1, \varGamma\gamma_1, \varGamma\delta_1)$ as $2 \times \frac{23}{32} \times (1.66 \times 2^{-3})^2 \approx 2^{-4}$ by their formula.

We first observe that the 3-round truncated differential is wrong, because of an error in the output difference of the round function of the second round. In the second round, only the S_3 S-box is active, and thus by the **P** permutation operation the correct output difference of the round function of the second round should be of the form $0x0S0T0U0V$, not the form $0x0X00NZ0Y$, where $S, V \in \{0, 4\}$ and $T, U \in \{0, 1\}$. Furthermore, we observe that the probability of the 3-round truncated differential is not correct. Biham et al. computed: (i) The probability that the second most significant bit of the output difference of the S_3 S-box of the second round is zero is $\frac{28}{64}$; and (ii) When the second most significant bit of the output difference of the S_3 S-box of the second round is 1, the probability that the second most significant bit of the output difference of the S_4 S-box of the third round is zero is $\frac{1}{2}$. As a result, they got that the differential has probability $\frac{28}{64} + (1 - \frac{28}{64}) \times \frac{1}{2} = \frac{46}{64}$. However, the probability given in (ii) is not correct. The S_4 S-box of the third round has an input difference $0x2$, and thus by the differential distribution table of the S_4 S-box (see [22]) we know that the probability that the second most significant bit of the output difference of the S-box is zero is $\frac{8+4+4+8}{64} = \frac{3}{8}$, not $\frac{1}{2}$. Therefore, the 3-round truncated differential should be $(\varDelta 0x0020000000000000 \rightarrow \varDelta 0x0S0T0U0V***JW***)$[4], where $S, V \in \{0, 4\}, T, U \in \{0, 1\}, J \in \{0, 2, 4, 6, 8, A, C, E\}, W \in \{0, 1, 2, 3, 4, 5, 6, 7\}$.

If we followed Biham et al.'s computation method, the correct 3-round truncated differential would have a probability of $\frac{28}{64} + (1 - \frac{28}{64}) \times \frac{3}{8} = \frac{83}{128}$ to make $T_L \odot \alpha_0 \times T_R \odot \beta_0 \oplus T_L \odot \gamma_0 \oplus T_R \odot \delta_0 = T_L^* \odot \alpha_0 \times T_R^* \odot \beta_0 \oplus T_L^* \odot \gamma_0 \oplus T_R^* \odot \delta_0$ (refer to Eq. (2) for its meaning). However, this is incorrect, as discussed earlier, and we need to check all the probabilities, instead of only a few of them. After a detailed analysis, we know that the differential always meets: (1) $T_L \odot \gamma_0 \oplus T_L^* \odot \gamma_0 = 0$; (2) $T_L \odot \alpha_0 \oplus T_L^* \odot \alpha_0 = 0$; (3) $T_R \odot \delta_0 \oplus T_R^* \odot \delta_0 = 0$; and (4) $T_R \odot 0x00010000 \oplus T_R^* \odot 0x00010000 = 0$. Thus, the equation to be predicted by the differential is simplified to $T_L \odot \alpha_0 \times (T_R \oplus T_R^*) \odot 0x00001000 = 0$, which is dependent of T_L and $T_R \oplus T_R^*$. By the **P** permutation operation of DES, we know that the value of $(T_R \oplus T_R^*) \odot 0x00001000$ is equal to the second most significant bit of the output difference of the S_4 S-box of the third round, which depends on T and can take 0 or 1. By the differential distribution table of the S_3 S-box, the probability of $T = 0$ is $\frac{4+12+8+4}{64} = \frac{7}{16}$. If $T = 0$ then $(T_R \oplus T_R^*) \odot 0x00001000 = 0$; and if $T = 1$ then by the differential distribution table of the S_4 S-box, $\Pr((T_R \oplus T_R^*) \odot 0x00001000 = 0) = \frac{8+4+4+8}{64} = \frac{3}{8}$. Hence, $\Pr((T_R \oplus T_R^*) \odot 0x00001000 = 0) = \frac{7}{16} + (1 - \frac{7}{16}) \times \frac{3}{8} = \frac{83}{128}$. If we assume that the plaintexts are randomly chosen, then $\Pr(T_L \odot \alpha_0 \times (T_R \oplus T_R^*) \odot 0x00001000 = 0) = \frac{83}{128} + \frac{1}{2} \times (1 - \frac{83}{128}) = \frac{211}{256}$. Finally from Theorem 3

[4] This is after the exchange of the left and right halves in the third round.

we learn that the bias of the differential-bilinear distinguisher $\langle \varDelta\alpha, \varGamma\alpha_0, \varGamma\beta_0, \varGamma\gamma_0, \varGamma\delta_0 \rangle \rightarrow (\varGamma\alpha_1, \varGamma\beta_1, \varGamma\gamma_1, \varGamma\delta_1)$ is $2\cdot|2\times\frac{211}{256}-1|\cdot(1.66\times 2^{-3})^2 \approx 2^{-4.16}$. As a result, the attack mentioned in [14] should require about $(\frac{2^{-4.16}}{2^{-4}})^{-2} \approx 1.25$ times of the originally required plaintexts.

Remarks. In [9] Biham et al. also applied the differential-bilinear analysis to break 8 rounds of the s^{5}DES [37] block cipher — a modified version of DES. Similarly, there are some flaws in the 8-round s^{5}DES attack, too.

9 Methodology for the Differential-Bilinear-Boomerang Analysis

In this section we deduce the probability of a differential-bilinear-boomerang distinguisher after reviewing Biham et al.'s differential-bilinear-boomerang analysis [9].

9.1 Review of the Differential-Bilinear-Boomerang Analysis

The differential-bilinear-boomerang analysis [9] is a combination of the boomerang analysis [18] with the differential-bilinear analysis. The boomerang analysis uses two differentials with larger probabilities on two different parts of a block cipher, instead of a single differential with a smaller probability on the entire cipher. For the sake of simplicity we assume the differential-bilinear analysis to be combined is that described in Section 8.1.

The differential-bilinear-boomerang analysis requires an attacker to build a differential-bilinear-boomerang distinguisher. Such a distinguisher treats $\mathbb{E}$ as a cascade of three sub-ciphers $\mathbb{E}_0$, $\mathbb{E}_1$ and $\mathbb{E}_2$, where $\mathbb{E} = \mathbb{E}_0 \circ \mathbb{E}_1 \circ \mathbb{E}_2$. It is made up of a differential $\varDelta\theta \rightarrow \varDelta\alpha$ with probability p_0 for $\mathbb{E}_0 \circ \mathbb{E}_1$, the set of the differentials for $\mathbb{E}_2$ that have a fixed output difference φ, denoted by $\{\varDelta\phi \rightarrow \varDelta\varphi | \Pr_{\mathbb{E}_2}(\varDelta\phi \rightarrow \varDelta\varphi) > 0, \phi \in \{0,1\}^n\}$ with $p_\phi = \Pr_{\mathbb{E}_2}(\varDelta\phi \rightarrow \varDelta\varphi)$, a bilinear approximation $(\varGamma\alpha_0, \varGamma\beta_0, \varGamma\gamma_0, \varGamma\delta_0) \rightarrow (\varGamma\alpha_1, \varGamma\beta_1, \varGamma\gamma_1, \varGamma\delta_1)$ with bias ϵ for $\mathbb{E}_0^{-1}$, (refer to Section 8.2 for its specifications), and a (truncated) differential $\varDelta\alpha \rightarrow \varDelta\beta$ for $\mathbb{E}_1^{-1}$ that meets the following equation with probability p:

$$T_L \odot \alpha_0 \times T_R \odot \beta_0 \oplus T_L \odot \gamma_0 \oplus T_R \odot \delta_0 = T_L^* \odot \alpha_0 \times T_R^* \odot \beta_0 \oplus T_L^* \odot \gamma_0 \oplus T_R^* \odot \delta_0, \tag{4}$$

where T_L, T_R are respectively the left and right halves of $\mathbb{E}_1^{-1}(Z)$ with Z chosen uniformly at random from $\{0,1\}^n$, and T_L^*, T_R^* are respectively the left and right halves of $\mathbb{E}_1^{-1}(Z \oplus \alpha)$.

Let P be a randomly chosen plaintext. The distinguisher focuses on the event expressed by Eq. (5).

$$\widetilde{P}_L \odot \alpha_1 \times \widetilde{P}_R \odot \beta_1 \oplus \widetilde{P}_L \odot \gamma_1 \oplus \widetilde{P}_R \odot \delta_1 = \widetilde{P}_L^* \odot \alpha_1 \times \widetilde{P}_R^* \odot \beta_1 \oplus \widetilde{P}_L^* \odot \gamma_1 \oplus \widetilde{P}_R^* \odot \delta_1, \tag{5}$$

where $\widetilde{P}_L, \widetilde{P}_R$ are respectively the left and right halves of $\mathbb{E}^{-1}(\mathbb{E}(P) \oplus \varphi)$, and $\widetilde{P}_L^*, \widetilde{P}_R^*$ are respectively the left and right halves of $\mathbb{E}^{-1}(\mathbb{E}(P \oplus \theta) \oplus \varphi)$.

Biham et al. showed that for a given value of α, the probability of $\mathbb{E}_2^{-1}(\mathbb{E}(P) \oplus \varphi) \oplus \mathbb{E}_2^{-1}(\mathbb{E}(P \oplus \theta) \oplus \varphi) = \alpha$ is $p_0 p_1^2$, where $p_1 = \sqrt{\sum_\phi p_\phi^2}$. Finally, by an analysis similar to the differential-bilinear analysis, they got that the bias of the differential-bilinear-boomerang distinguisher is $|\Pr(Eq.\ (5)\ holds) - \frac{1}{2}| = 2p_0 p_1^2 p \epsilon^2$. Thus, if $2p_0 p_1^2 p \epsilon^2$ is sufficiently large, the distinguisher can be used to distinguish $\mathbb{E}$ from a random function.

In [9] Biham et al. described some theoretical advantages of the differential-bilinear-boomerang analysis. There have been no published applications for the technique. Biham et al. hinted (and Dunkelman explicitly stated in [14]) that with small modifications the differential-bilinear-boomerang analysis covers the combinations of the boomerang analysis with linear, differential-linear and bilinear cryptanalysis, as they are special cases of the differential-bilinear analysis.

9.2 New Methodology for Computing the Probability of a Differential-Bilinear-Boomerang Distinguisher

First, Biham et al.'s methodology is not correct in some situations. Consider an intuitive situation such that: (1) The differential $\varDelta\theta \rightarrow \varDelta\alpha$ for $\mathbb{E}_0 \circ \mathbb{E}_1$ has probability 1; (2) There is one differential $\varDelta\phi \rightarrow \varDelta\varphi$ for $\mathbb{E}_2$ that has probability 1; and (3) The differential $\varDelta\alpha \rightarrow \varDelta\beta$ for $\mathbb{E}_1^{-1}$ makes Eq. (4) hold with probability $\frac{1}{2}$ and makes it not hold with probability $\frac{1}{2}$. Under the situation, it is certain that the input difference for $\mathbb{E}_1^{-1}$ is α, and thus the situation is simplified to analyse the bias of a differential-bilinear distinguisher. After a similar analysis, we know that the differential-bilinear-boomerang distinguisher has a bias of 0. However, Biham et al.'s formula suggests that it is $2 \times 1 \times 1 \times \frac{1}{2} \times \epsilon^2 = \epsilon^2$, which is wrong.

For simplify, we denote by $\langle \varDelta\theta, \varDelta\varphi, \varGamma\alpha_0, \varGamma\beta_0, \varGamma\gamma_0, \varGamma\delta_0\rangle \rightarrow (\varGamma\alpha_1, \varGamma\beta_1, \varGamma\gamma_1, \varGamma\delta_1)$ a differential-bilinear-boomerang distinguisher that focuses on the event expressed by Eq. (5).

Biham et al. considered only one input difference (i.e., α) for $\mathbb{E}_1^{-1}$, however, this is not sufficient in most situations, and there may exist many possible input differences $\{\alpha| \Pr(\mathbb{E}_2^{-1}(\mathbb{E}(P)\oplus\varphi)\oplus\mathbb{E}_2^{-1}(\mathbb{E}(P\oplus\theta)\oplus\varphi) = \alpha) > 0, \alpha \in \{0,1\}^n\}$; we label the set $\mathcal{S}$, and assume $\widehat{p}_\alpha = \Pr(\mathbb{E}_2^{-1}(\mathbb{E}(P) \oplus \varphi) \oplus \mathbb{E}_2^{-1}(\mathbb{E}(P \oplus \theta) \oplus \varphi) = \alpha)$; and given α, Eq. (4) holds with probability p_α. Then by applying Theorem 3, we get that the differential-bilinear-boomerang distinguisher has a probability $\sum_{\alpha\in\mathcal{S}}[\widehat{p}_\alpha \times (\frac{1}{2} + 2(2p_\alpha - 1)\epsilon^2)] = \frac{1}{2} + 2\epsilon^2 \sum_{\alpha\in\mathcal{S}}[\widehat{p}_\alpha(2p_\alpha - 1)]$. Hence, we have the following result.

Theorem 4. *An n-bit block cipher $\mathbb{E}$ is represented as a cascade of three sub-ciphers $\mathbb{E}_0$, $\mathbb{E}_1$ and $\mathbb{E}_2$, where $\mathbb{E} = \mathbb{E}_0 \circ \mathbb{E}_1 \circ \mathbb{E}_2$. Let P be a randomly chosen plaintext. If θ $(\neq 0)$ is an input difference for $\mathbb{E}_0$, φ $(\neq 0)$ is an output difference for $\mathbb{E}_2$, $(\varGamma\alpha_0, \varGamma\beta_0, \varGamma\gamma_0, \varGamma\delta_0) \rightarrow (\varGamma\alpha_1, \varGamma\beta_1, \varGamma\gamma_1, \varGamma\delta_1)$ is a bilinear approximation with bias ϵ for $\mathbb{E}_0^{-1}$, let $\mathcal{S} = \{\alpha| \Pr(\mathbb{E}_2^{-1}(\mathbb{E}(P) \oplus\varphi)\oplus\mathbb{E}_2^{-1}(\mathbb{E}(P\oplus\theta)\oplus\varphi) = \alpha) > 0, \alpha \in \{0,1\}^n\}$ be the set of possible input differences for $\mathbb{E}_1^{-1}$ with $\widehat{p}_\alpha = \Pr(\mathbb{E}_2^{-1}(\mathbb{E}(P) \oplus \varphi) \oplus \mathbb{E}_2^{-1}(\mathbb{E}(P \oplus \theta) \oplus \varphi) = \alpha)$, and Eq. (4) holds for a given α with probability p_α, then the probability of the differential-bilinear-boomerang distinguisher $\langle \varDelta\theta, \varDelta\varphi, \varGamma\alpha_0, \varGamma\beta_0, \varGamma\gamma_0, \varGamma\delta_0\rangle \rightarrow (\varGamma\alpha_1, \varGamma\beta_1, \varGamma\gamma_1, \varGamma\delta_1)$ is $\Pr(Eq.\ (5)\ holds) = \frac{1}{2} + 2\epsilon^2 \sum_{\alpha\in\mathcal{S}}[\widehat{p}_\alpha(2p_\alpha - 1)]$.*

Therefore, the bias of the differential-bilinear-boomerang distinguisher $\langle \varDelta\theta,\ \varDelta\varphi, \varGamma\alpha_0, \varGamma\beta_0, \varGamma\gamma_0, \varGamma\delta_0\rangle \rightarrow (\varGamma\alpha_1, \varGamma\beta_1, \varGamma\gamma_1,\ \varGamma\delta_1)$ is $|\Pr(Eq.\ (5)\ holds) - \frac{1}{2}| = 2\epsilon^2|\sum_{\alpha\in\mathcal{S}}[\widehat{p}_\alpha(2p_\alpha - 1)]|$.

9.3 Implications

Biham et al. used only one input difference for $\mathbb{E}_1^{-1}$. However, this is not sufficient in most situations, and we should try to use as many input differences as possible, which is much tougher than originally thought in practice. Biham et al.'s methodology does not have the generality to describe the differential-bilinear-boomerang analysis. Other implications are similar to those described in Section 3.5.

10 Conclusions

In this paper we have shown that Biham et al.'s methodologies for computing the probabilities of a differential-linear distinguisher, a high-order differential-linear distinguisher, a differential-bilinear distinguisher and a differential-bilinear-boomerang distinguisher do not have the generality to describe the cryptanalytic techniques, and have given general methodologies for computing the probabilities under the general assumptions. Using the new methodologies, we have presented differential-linear attacks on 13-round DES, 10-round CTC2 with a 255-bit block size and key, and 12-round Serpent with a 256-bit key, and have corrected Biham et al.'s differential-bilinear attack on 8-round DES. Like most cryptanalytic results on block ciphers, the presented attacks are theoretical in the sense of the magnitudes of the attack complexities.

Most recently, we note that Liu et al. [38] described a new extension of differential-linear cryptanalysis, called differential-multiple linear cryptanalysis, which uses more than one linear approximations. They computed the probability of a so-called differential-multiple linear distinguisher in a similar way to Biham et al.'s enhanced approach. As a result, we learn that this probability formula does not have the generality, either. The general formula can be obtained as for a differential-linear distinguisher in Theorem 1.

Acknowledgments

The author is very grateful to the anonymous referees for their comments.

References

1. Biham, E., Shamir, A.: Differential cryptanalysis of DES-like cryptosystems. In: Menezes, A., Vanstone, S.A. (eds.) CRYPTO 1990. LNCS, vol. 537, pp. 2–21. Springer, Heidelberg (1990)

2. Matsui, M., Yamagishi, A.: A new method for known plaintext attack of FEAL cipher. In: Rueppel, R.A. (ed.) EUROCRYPT 1992. LNCS, vol. 658, pp. 81–91. Springer, Heidelberg (1993)
3. National Institute of Standards and Technology (NIST), Data Encryption Standard (DES), FIPS-46 (1977)
4. Biham, E., Shamir, A.: Differential cryptanalysis of the full 16-round DES. In: Brickell, E.F. (ed.) CRYPTO 1992. LNCS, vol. 740, pp. 487–496. Springer, Heidelberg (1993)
5. Matsui, M.: Linear cryptanalysis method for DES cipher. In: Helleseth, T. (ed.) EUROCRYPT 1993. LNCS, vol. 765, pp. 386–397. Springer, Heidelberg (1994)
6. Langford, S.K., Hellman, M.E.: Differential-linear cryptanalysis. In: Desmedt, Y. (ed.) CRYPTO 1994. LNCS, vol. 839, pp. 17–25. Springer, Heidelberg (1994)
7. Knudsen, L.R.: Trucated and higher order differentials. In: Preneel, B. (ed.) FSE 1994. LNCS, vol. 1008, pp. 196–211. Springer, Heidelberg (1995)
8. Biham, E., Dunkelman, O., Keller, N.: Enhancing differential-linear cryptanalysis. In: Zheng, Y. (ed.) ASIACRYPT 2002. LNCS, vol. 2501, pp. 254–266. Springer, Heidelberg (2002)
9. Biham, E., Dunkelman, O., Keller, N.: New combined attacks on block ciphers. In: Gilbert, H., Handschuh, H. (eds.) FSE 2005. LNCS, vol. 3557, pp. 126–144. Springer, Heidelberg (2005)
10. Biham, E., Dunkelman, O., Keller, N.: Differential-linear cryptanalysis of Serpent. In: Johansson, T. (ed.) FSE 2003. LNCS, vol. 2887, pp. 9–21. Springer, Heidelberg (2003)
11. Biham, E., Dunkelman, O., Keller, N.: A new attack on 6-round IDEA. In: Biryukov, A. (ed.) FSE 2007. LNCS, vol. 4593, pp. 211–224. Springer, Heidelberg (2007)
12. Dunkelman, O., Indesteege, S., Keller, N.: A differential-linear attack on 12-round Serpent. In: Chowdhury, D.R., Rijmen, V., Das, A. (eds.) INDOCRYPT 2008. LNCS, vol. 5365, pp. 308–321. Springer, Heidelberg (2008)
13. Dunkelman, O., Keller, N.: Cryptanalysis of CTC2. In: Fischlin, M. (ed.) CT-RSA 2009. LNCS, vol. 5473, pp. 226–239. Springer, Heidelberg (2009)
14. Dunkelman, O.: Techniques for cryptanalysis of block ciphers. PhD thesis, the Technion — Israel Institute of Technology, 2006.
15. Courtois, N.T.: CTC2 and fast algebraic attacks on block ciphers revisited. IACR ePrint report 2007/152 (2007)
16. Biham, E., Dunkelman, O., Keller, N.: The rectangle attack — rectangling the Serpent. In: Pfitzmann, B. (ed.) EUROCRYPT 2001. LNCS, vol. 2045, pp. 340–357. Springer, Heidelberg (2001)
17. Biham, E., Dunkelman, O., Keller, N.: Linear cryptanalysis of reduced round Serpent. In: Matsui, M. (ed.) FSE 2001. LNCS, vol. 2355, pp. 16–27. Springer, Heidelberg (2002)
18. Wagner, D.: The boomerang attack. In: Knudsen, L.R. (ed.) FSE 1999. LNCS, vol. 1636, pp. 156–170. Springer, Heidelberg (1999)
19. Kelsey, J., Kohno, T., Schneier, B.: Amplified boomerang attacks against reduced-round MARS and Serpent. In: Schneier, B. (ed.) FSE 2000. LNCS, vol. 1978, pp. 75–93. Springer, Heidelberg (2001)
20. Biham, E., Dunkelman, O., Keller, N.: New results on boomerang and rectangle attacks. In: Daemen, J., Rijmen, V. (eds.) FSE 2002. LNCS, vol. 2365, pp. 1–16. Springer, Heidelberg (2002)
21. Handschuh, H., Naccache, D.: SHACAL. In: Proceedings of the First Open NESSIE Workshop (2000)
22. Biham, E., Shamir, A.: Differential cryptanalysis of DES-like cryptosystems. Journal of Cryptology 4(1), 3–72 (1991)
23. Matsui, M.: The first experimental cryptanalysis of the Data Encryption Standard. In: Desmedt, Y.G. (ed.) CRYPTO 1994. LNCS, vol. 839, pp. 1–11. Springer, Heidelberg (1994)
24. Knudsen, L.R., Mathiassen, J.E.: A chosen-plaintext linear attack on DES. In: Schneier, B. (ed.) FSE 2000. LNCS, vol. 1978, pp. 262–272. Springer, Heidelberg (2001)
25. Hawkes, P.: Differential-linear weak key classes of IDEA. In: Nyberg, K. (ed.) EUROCRYPT 1998. LNCS, vol. 1403, pp. 112–126. Springer, Heidelberg (1998)
26. Biham, E.: New types of cryptanalytic attacks using related keys. In: Helleseth, T. (ed.) EUROCRYPT 1993. LNCS, vol. 765, pp. 398–409. Springer, Heidelberg (1993)
27. Knudsen, L.R.: Cryptanalysis of LOKI91. In: Seberry, J., Zheng, Y. (eds.) ASIACRYPT 1992. LNCS, vol. 718, pp. 196–208. Springer, Heidelberg (1993)
28. Kim, J.: Combined differential, linear and related-key attacks on block ciphers and MAC algorithms. PhD thesis, Katholieke Universiteit Leuven, 2006.
29. Selçuk, A.A.: On probability of success in linear and differential cryptanalysis. Journal of Cryptology 21(1), 131–147 (2008)
30. Courtois, N.T.: Feistel schemes and bi-linear cryptanalysis. In: Franklin, M.K. (ed.) CRYPTO 2004. LNCS, vol. 3152, pp. 23–40. Springer, Heidelberg (2004)
31. Anderson, R., Biham, E., Knudsen, L.R.: Serpent: a proposal for the Advanced Encryption Standard, NIST AES proposal (1998)
32. Daemen, J., Rijmen, V.: AES proposal: Rijndael, Proceedings of The First Advanced Encryption Standard Candidate Conference, NIST (1998)
33. Wang, X.Y., Hui, L.C.K., Chow, K.P., Chong, C.F., Tsang, W.W., Chan, H.W.: The differential cryptanalysis of an AES finalist — Serpent. Technical report TR-2000-04, Department of Computer Science & Information Systems, The University of Hong Kong.

34. Lai, X.: Higher order derivatives and differential cryptanalysis. In: Communications and Cryptography, pages 227–233, 1994. Academic Publishers.
35. Shimizu, A., Miyaguchi, S.: Fast Data Encipherment Algorithm FEAL. In: Günther, C.G. (ed.) EUROCRYPT 1987. LNCS, vol. 304, pp. 267–278. Springer, Heidelberg (1987)
36. Lai, X., Massey, J.L.: A proposal for a new block cipher encryption standard. In: Damgård, I. (ed.) EUROCRYPT 1990. LNCS, vol. 473, pp. 389–404. Springer, Heidelberg (1991)
37. Kim, K., Lee, S., Park, S., Lee, D.: How to strengthen DES against two robust attacks. In: Proceedings of the 1995 Korea-Japan Joint Workshop on Information Security and Cryptology (1995)
38. Liu, Z., Gu, D., Zhang, J., Li, W.: Differential-multiple linear cryptanalysis. In: Feng, B., Yung, M. (eds.) INSCRYPT 2009. LNCS, to appear.

A A 9-Round Differential-Linear Distinguisher with Bias $2^{-59.01}$ of Serpent

The 9-round distinguisher has been introduced in Section 3.7, and below we will compute its bias. The input difference $\Delta\alpha$ makes only one active S-box among the 32 S_5^{-1} S-boxes of Round 13, which has 7 possible output differences: $\{\omega|\omega = 0x1, 0x4, 0x6, 0x8, 0xC, 0xD, 0xE\}$; the probabilities for these output differences are given in the second column of Table 9, and the differential distribution tables of the eight S-boxes are presented in [33]. We write ω as $d||c||b||a$ in binary notation, where $a, b, c, d \in \{0, 1\}$. It is easy to see that the input mask $\Gamma\gamma$ concerns a total of 16 bits of the output differences of seven S_3^{-1} S-boxes in Round 11: (i) The least significant two bits and the most significant bit of the output difference of S-box 28, and we denote by y_0 the XOR of the three bits; (ii) The least significant two bits and the most significant bit of the output difference of S-box 23, and we denote by y_1 the XOR of the three bits; (iii) The least significant two bits of the output difference of S-box 18, and we denote by y_2 the XOR of the two bits; (iv) The least significant two bits and the most significant bit of the output difference of S-box 13, and we denote by y_3 the XOR of the three bits; (v) The second least significant bit of the output difference of S-box 11, and we label it y_4; (vi) The most significant three bits of the output difference of S-box 8, and we denote by y_5 the XOR of the three bits; and (vii) The least significant bit of the output difference of S-box 1, and we label it y_6.

We now focus on ten S_4^{-1} S-boxes in Round 12. S-box 31 has an input difference $000a$ in binary notation, and we denote the most significant bit, the second least significant bit and the least significant bit of its output difference by x_0, x_1, x_2, respectively. S-box 24 has an input difference $bcb0$ in binary notation, and we denote the most significant bit and the second least significant bit of its output difference by x_3, x_4, respectively. S-box 21 has an input difference $0b00$ in binary notation, and we denote the most significant bit, the second most significant bit and the second least significant bit of its output difference by x_5, x_6, x_7, respectively. S-box 20 has an input difference $(b \oplus d)000$ in binary notation, and we denote the most significant bit, the second least significant bit and the least significant bit of its output difference by x_8, x_9, x_{10}, respectively. S-box 16 has an input difference $00b0$ in binary notation, and we denote by $x_{11}||x_{12}||x_{13}||x_{14}$ the four-bit output difference, where $x_{11}, x_{12}, x_{13}, x_{14} \in \{0, 1\}$. S-box 15 has an input difference $a000$ in binary notation, and we denote the most significant bit, the second least significant bit and the least significant bit of its output difference by x_{15}, x_{16}, x_{17}, respectively. S-box 14 has an input difference $0d00$ in binary notation, and we denote the most significant bit, the second least significant bit and the least significant bit of its output difference by x_{18}, x_{19}, x_{20}, respectively. S-box 12 has an input difference $00a0$ in binary notation, and we denote by x_{21} the second least significant bit of its output difference. S-box 10 has an input difference $d000$ in binary notation, and we denote the most significant bit, the second least significant bit and the least significant bit of its output difference by x_{22}, x_{23}, x_{24}, respectively. S-box 4 has an input difference $000(b \oplus d)$ in binary notation, and we denote by $x_{25}||x_{26}||x_{27}||x_{28}$ the four-bit output difference, where $x_{25}, x_{26}, x_{27}, x_{28} \in \{0, 1\}$.

As a result, we get the input differences of the concerned seven S_3^{-1} S-boxes of Round 11: (1) The input difference for S-box 28 is $x_7||(x_0 \oplus x_4 \oplus x_6)||x_7||x_{20}$; (2) The input difference for S-box 23 is $(x_8 \oplus x_9 \oplus x_{13})||x_{12}||(x_4 \oplus x_{13})||(x_{25} \oplus x_{27})$; (3) The input difference for S-box 18 is $(x_{10} \oplus x_{15} \oplus x_{16})||(x_5 \oplus x_{19})||0||(x_0 \oplus x_1 \oplus x_{28})$; (4) The input difference for S-box 13 is $(x_8 \oplus x_{17} \oplus x_{22} \oplus x_{23})||x_{11}||x_{19}||x_2$; (5) The input difference for S-box 11 is $x_{27}||(x_{18} \oplus x_{26})||(x_{14} \oplus x_{21} \oplus x_{27})||(x_3 \oplus x_4)$; (6) The input difference for S-box 8 is $(x_{15} \oplus x_{24})||x_{27}||0||(x_5 \oplus x_7)$; and (7) The input difference for S-box 1 is $0||x_{25}||0||(x_{18} \oplus x_{19})$.

By the differential distribution table of S_4^{-1}, we similarly get all the possible values for $(x_0||x_1||x_3)$, $(x_3||x_4)$, $(x_5||x_6||x_7)$, $(x_8||x_9||x_{10})$, $(x_{11}||x_{12}||x_{13}||x_{14})$, $(x_{15}||x_{16}||x_{17})$, $(x_{18}||x_{19}||x_{20})$, x_{21}, $(x_{22}||x_{23}||x_{24})$, $(x_{25}||x_{26}||x_{27}||x_{28})$ as well as their probabilities. By the differential distribution table of S_3^{-1}, we similarly get the probability that for every possible input difference, the XOR of the concerned bit(s) of the output difference of each concerned S_3^{-1} S-box in Round 11 is 0 or 1. For each difference ω, we denote by β_ω the output difference(s) of the 2.5-round Serpent

Table 9. Probabilities for the seven output differences in $\{\omega\}$

Difference (ω)	$\Pr_{S_5^{-1}}(\Delta 0x9 \rightarrow \Delta\omega)$	$\Pr(\Delta\beta_\omega \odot \Gamma\gamma = 0 \vert \Delta 0x9 \rightarrow \Delta\omega)$
$0x1$	2^{-3}	0.4996337890625
$0x4$	2^{-2}	0.53125
$0x6$	2^{-3}	0.49991488456726074
$0x8$	2^{-3}	0.5
$0xC$	2^{-3}	0.5
$0xD$	2^{-3}	0.5
$0xE$	2^{-3}	0.5

in the decryption direction. Subsequently, we compute the probability the XOR of the concerned 16 bits of β_ω is zero (i.e., $\Pr(\Delta\beta_\omega \odot \Gamma\gamma = 0|\Delta 0x9 \rightarrow \Delta\omega)$) by performing a program over all the possible (truncated) differential characteristics. These probabilities are given in the third column of Table 9.

Therefore, the probability of the 9-round differential-linear distinguisher is $\frac{1}{2} + [2 \times \sum_\omega \Pr_{S_5^{-1}}(\Delta 0x9 \rightarrow \Delta\omega) \times \Pr(\Delta\beta_\omega \odot \Gamma\gamma = 0|\Delta 0x9 \rightarrow \Delta\omega) - 1] \times (2^{-27})^2 \approx \frac{1}{2} + 2 \times 2^{-6.01} \times 2^{-54} = \frac{1}{2} + 2^{-59.01}$. Thus, the 9-round differential-linear distinguisher has a bias of $2^{-59.01}$.

B Differential-Linear Attack on 10-Round DES

The 8-round distinguisher $\langle \Delta\widehat{\alpha}, \Gamma\gamma\rangle \rightarrow \Gamma\delta$ enables us to construct a differential-linear attack breaking 10 rounds of DES. We assume the attacked rounds are the first ten rounds from Rounds 1 to 10. The attack procedure is as follows.

1. Choose $2^{24.66}$ structures S_i, $(i = 1, 2, \cdots, 2^{24.66})$, where a structure is defined to be a set of 2^4 plaintexts $P_{i,j}$ with bits (9,17,23, 31) of the left half taking all the possible values, bit (2) of the right half fixed to 0 and the other 59 bits fixed, $(j = 1, 2, \cdots, 2^4)$. In a chosen-plaintext attack scenario, obtain all the ciphertexts for the 2^4 plaintexts in each of the $2^{24.66}$ structures; we denote by $C_{i,j}$ the ciphertext for plaintext $P_{i,j}$.
2. Choose $2^{24.66}$ structures $\widehat{S}_i$, $(i = 1, \cdots, 2^{24.66})$, where a structure $\widehat{S}_i$ is obtained by setting 1 to bit (2) of the right half of all the plaintexts $P_{i,j}$ in S_i. In a chosen-plaintext attack scenario, obtain all the ciphertexts for the 2^4 plaintexts in each $\widehat{S}_i$.
3. Initialize 2^{12} counters to zero, which correspond to all the possible values of the 12-bit subkey $(K_{1,1}, K_{10,1})$.
4. Guess the 6-bit subkey $K_{1,1}$, and do as follows.
   (a) Partially encrypt every plaintext $P_{i,j}$ with the guessed $K_{1,1}$ to get its intermediate value immediately after Round 1; we denote it by $\varepsilon_{i,j}$.
   (b) Partially decrypt $\varepsilon_{i,j} \oplus 0x4000000000000000$ with the guessed $K_{1,1}$ to get its plaintext, and find the plaintext in $\widehat{S}_i$; we denote it by $\widehat{P}_{i,j}$, and denote by $\widehat{C}_{i,j}$ the corresponding ciphertext for $\widehat{P}_{i,j}$. Store $(C_{i,j}, \widehat{C}_{i,j})$ in a table.
5. Guess the 6-bit subkey $K_{10,1}$, and do as follows for every ciphertext pair $(C_{i,j}, \widehat{C}_{i,j})$.
   (a) Partially decrypt $C_{i,j}$ and $\widehat{C}_{i,j}$ with $K_{10,1}$ to get bit (17) of the left half of their intermediate values immediately before Round 10.
   (b) Check whether the XOR of the five bits for $C_{i,j}$ — bit (17) of the left half and bits (3,8,14,25) of the right half of its intermediate value immediately before Round 10 — is equal to the XOR of the corresponding five bits for $\widehat{C}_{i,j}$. If yes, increase 1 to the counter corresponding to the guessed $(K_{1,1}, K_{10,1})$.
6. Output the guess for $(K_{1,1}, K_{10,1})$ with the highest deviation from $2^{27.66}$.

The attack requires $2^{29.66}$ chosen plaintexts, and has a time complexity of $2 \times 2^{28.66} + 2 \times 2^{28.66} \times 2^6 \times \frac{1}{80} + 2 \times 2^{28.66} \times 2^{12} \times \frac{1}{80} \approx 2^{35.36}$ 10-round DES encryptions. There are $2^{28.66}$ plaintext pairs $(P_{i,j}, \widehat{P}_{i,j})$ for a guess of $(K_{1,1}, K_{10,1})$. Following the results in [29], we learn that the attack has a success probability of about 97%. The remaining key bits can be found by exhaustive search.