On Multidimensional Linear Cryptanalysis

Phuong Ha Nguyen, Lei Wei, Huaxiong Wang, and San Ling

Division of Mathematical Sciences,
School of Physical and Mathematical Sciences,
Nanyang Technological University, Singapore
{ng0007ha,weil0005,hxwang,lingsan}@ntu.edu.sg

Abstract. Matsui's Algorithms 1 and 2 with multiple approximations have been studied over 16 years. In CRYPTO'04, Biryukov *et al.* proposed a formal framework based on m statistically independent approximations. Started by Hermelin *et al.* in ACISP'08, a different approach was taken by studying m-dimensional combined approximations from m base approximations. Known as multidimensional linear cryptanalysis, the requirement for statistical independence is relaxed. In this paper we study the multidimensional Alg. 1 of Hermelin *et al.*. We derive the formula for N, the number of samples required for the attack and we improve the algorithm by reducing time complexity of the distillation phase from $2^m N$ to $2m2^m + mN$, and that of the analysis phase from 2^{2m} to $3m2^m$. We apply the results on 4- and 9-round Serpent and show that Hermelin *et al.* actually provided a formal model for the hypothesis of Biryukov *et al.* in practice, and this model is now much more practical with our improvements.

1 Introduction

Linear cryptanalysis [12] was formally introduced in 1993 by Matsui, who suggested 2 algorithms to exploit linear approximations of block ciphers. Consider a block cipher $E_k(\cdot)$ with a linear approximation: $g = uX \oplus vY \oplus cK$ and $Pr(g = 0) = 1/2 + \epsilon$, where u, v, and c are the selection patterns for the plaintext, ciphertext and the extended key, respectively, X is the plaintext and Y is ciphertext, K is extended key of secret key k, and ϵ is the bias of linear approximation.

Algorithm 1 in [12] is a known-plaintext attack and it requires a pool of sufficiently many random plaintext-ciphertext pairs. If successful, the parity cK can be recovered and N – the number of random samples as (X, Y) pairs needed – is proportional to c/ϵ^2, where c is a constant that depends on the success probability. In 1994 Matsui provided the first experimental cryptanalysis of DES [13], using two linear approximations derived from the best 14-round expression.

From then researchers started the quest for better attacks by using multiple linear approximations to improve the basic form of Alg. 1 and 2 in [12]. In the same year, Kaliski and Robshaw [11] combined m linear approximations $g_1, ..., g_m$, with biases of $\epsilon_1, \ldots, \epsilon_m$ respectively. The linear approximations are required to have the same selection pattern c for the extended key. N can achieve

R. Steinfeld and P. Hawkes (Eds.): ACISP 2010, LNCS 6168, pp. 37–52, 2010.

an m-fold reduction while keeping the same success rate, with N proportional to $1/\sum_{i=1}^{m} \epsilon_i^2$. Due to the restriction on the key mask, it recovers at most 1-bit parity $- c \cdot K$. Intuitively speaking, asking 1 parity bit from m approximations simultaneously would require much fewer samples than from a single approximation, under the same success rate.

In 2004 Biryukov *et al.* [2] show a statistical framework for using multiple approximations in both Alg 1 and Alg 2. In their generalization to Alg. 1, m statistically independent linear approximation $g_1, \ldots, g_m$ are used, with biases of $\epsilon_1, \ldots, \epsilon_m$ respectively. The attack is able to recover at most m parity bits if successful, with N proportional to $1/\sum_{i=1}^{m} \epsilon_i^2$. In practice, m' linearly dependent masks are used with $m' > m$. It is expected that the performance in this case is strictly better than that of m independent approximations, although no explicit estimation of N was given. They confirmed the reduction in data complexity by using $m' = 86$ approximations for 8-round DES, the 86 approximations only give 10 linearly independent key masks. In 2007, this reduction in N by using multiple approximations is further confirmed by experiments of Collard *et al.* [7], with $m' = 64$ for 4-round Serpent from 10 linearly independent text masks. Collard *et al.* also observed that the gain increases 8 times faster with 64 approximations than with 10 approximations. Both [2] and [7] did not analyze why such an advantage is present. Some intuitive guesses were given by Collard *et al.* [7], referring to linear hulls, error correcting codes, etc. They also noticed that in practice, m and m' cannot be too large due to computation limits.

Significant reduction in data complexity can be achieved by the methods in [2]. However, in practice, it is generally not easy to verify whether the m approximations are statistically independent. Instead, linear independence is used as a criterion for the m approximations, in both [2] and [7]. In the meantime, it is also helpful to doubt whether there is a better attack if the m approximations are linearly independent in text masks but statistically correlated, as shown by the experimental results in [2] and [7]. Notably, these experiments work with linearly dependent text masks.

Hermelin *et al.* [10] introduced a multidimensional framework, in which Matsui's Alg. 1 is generalized to m-dimensions to exploit correlations between the m approximations, to achieve a higher capacity. Experiments on 4-round Serpent have shown that this method reduces N compared with a similar attack in [7]. In this framework, the requirement for statistical independence of approximations is relieved. Instead, the approximations only need to have linearly independent text masks. This resembles the experiment scenarios in [2] and [7]. In fact, in [10] it is shown clearly that the statistical independence assumption of [2] does not hold.

In [10], the formula for N is derived as the amount of data needed to tackle the $|\mathcal{Z}|$-ary hypothesis testing problem, where $\mathcal{Z}$ is the set of key classes. It does not reflect how it depends on the approximations $g_1, \cdots, g_m$ and it is not efficiently computable. The reduction in N compared with [2] was observed empirically, rather than theoretically. Our first contribution is the provision with proof of a much simpler theoretical formula for the number of samples required, to complement this

theoretical framework in [10]. This formula gives us insights on how much can be achieved with multidimensional linear cryptanalysis. We can now easily estimate N given $g_1, \ldots, g_m$, hence the attack complexity. The simplicity of the formula eases cryptanalysis greatly since to compute N with the original formula in [10] (for N_{key}) requires a lot of computation when m is large.

A major obstacle for the method in [10] to be useful is that the number of approximations m is much limited due to complexity bottlenecks in the distillation phase ($2^m N$) and the analysis phase (2^{2m}) of the online stage. This limitation hinders the application of multidimensional Alg. 1 in practice when more approximations have to be used. Our second contribution is that we present a method (Method-A) for the distillation phase to speed up the computation for distributions to $2m2^m + mN$, using ideas from [6], and we develop Method-B to the analysis phase which reduces the complexity to $3m2^m$. We arrive at an improved algorithm for the multidimensional generalization of Matsui's Alg. 1 in [10], better than previous generalizations to a number of degrees. Most importantly, the algorithm is practical, with strong support from theoretical work in [10]. We show applications to 4- and 9-round Serpent as examples.

The paper is organized as follows. Section 2 contains notations and some basic notions. Section 3 is about the statistical model and algorithm in [10]. We show briefly how the distribution of an m-dimensional vectorial Boolean function g can be constructed from m given approximations $g_1, \ldots, g_m$ as Boolean functions. Section 4 contains our contributions. After the proof for N, we analyze the algorithm of [10] step-wise, to introduce our improvements, followed by the descriptions for each improvement. Section 5 describes the improved algorithm in detail. Section 6 gives the application results on Serpent and comparisons with previous cryptanalysis results [1] and [4]. In Section 7, we conclude and discuss the implication of our work.

2 Notations and Background

We follow the notations used in [10]. Denote the space of m-dimensional binary vectors by V_m or $V_m := GF(2)^m$. The inner product of 2 vectors $a = (a_1, \ldots, a_m), b = (b_1, \ldots, b_m), a, b \in V_m$ is $ab = \bigoplus_{i=1}^{m} a_i b_i$.

The function $f : V_m \to V_1$ is called a Boolean function and $f : V_n \to V_m, f = (f_1, \ldots, f_m)$ is called a vectorial Boolean function, where each f_i is a Boolean function for all $i = 1, \cdots, m$.

Let X be a random variable (r.v.) in V_m. Let $p_\eta = Pr(X = \eta)$, with $\eta \in V_m$. Then $p = (p_0, p_1, \ldots, p_{2^m-1})$ is the probability distribution (p.d) of r.v. X. If we associate with a vectorial Boolean function $f : V_n \to V_m$ an r.v. $Y := f(X)$, where X is uniformly distributed in V_m, then the p.d. of Y is $p(f) := (p_0(f), \ldots, p_{2^m-1}(f))$ where $p_\eta(f) = Pr(f(X) = \eta)$, for all $\eta \in V_m$. Two Boolean functions f and g are called statistically independent if their associated r.v.'s $f(X)$ and $g(Y)$ are statistically independent, with X, Y uniform in V_n.

The correlation between a binary r.v. X and 0 is $\rho = Pr(X = 0) - Pr(X = 1) = 2\epsilon$, where ϵ is the bias of the r.v. X. Let $g : V_m \to V_1$ be a Boolean function. Its correlation with 0 is defined as

$$\rho = 2^{-m}(\#\{\eta \in V_m | g(\eta) = 0\}) - \#\{\eta \in V_m | g(\eta) = 1\}) = 2Pr(g(X) = 0) - 1,$$

where X is uniformly distributed in V_m.

Definition 1. *Let $p = (p_0, \ldots, p_M)$ and $q = (q_0, \ldots, q_M)$ be two p.d.'s. Then their (mutual) capacity is*

$$C(p||q) = \sum_{\eta=0}^{M} \frac{(p_\eta - q_\eta)^2}{q_\eta}. \tag{1}$$

Definition 2. *The relative entropy or the Kullback-Leibler (KL) distance between two distributions $p = (p_0, \ldots, p_M)$ and $q = (q_0, \ldots, q_M)$ is defined as*

$$D(q||p) = \sum_{\eta=0}^{M} q_\eta \log \frac{q_\eta}{p_\eta}.$$

In [6], Collard *et al.* presented the following theorems concerning circulant matrices.

Theorem 1. *A circulant $\mathbf{S}$ of level k and type $(m, n, o, \ldots, r)$ is diagonalizable by the unitary matrix $\mathbf{F} = \mathbf{F}_m \otimes \mathbf{F}_n \otimes \mathbf{F}_o \otimes \cdots \otimes \mathbf{F}_r$*

$$\mathbf{S} = \mathbf{F}^* diag(\lambda)\mathbf{F},$$

where λ is the vector of eigenvalues of $\mathbf{S}$, the symbol $\otimes$ is the Kronecker product and $\mathbf{F}_n$ is the Fourier matrix of size $n \times n$ defined by:

$$\mathbf{F}_n(i, j) = \frac{1}{\sqrt{n}}\omega^{ij}, \ (0 \le i, j \le n - 1)$$

with

$$\omega = e^{\frac{2\pi\sqrt{-1}}{n}}.$$

Theorem 2. *The eigenvalues vector λ of a circulant matrix $\mathbf{S}$ of level k and type $(m, n, o, \ldots, r)$ can be computed with the following matrix-vector product:*

$$\lambda = \mathbf{FS}(:, 1)\sqrt{mno \ldots r}$$

where $\mathbf{S}(:, 1)$ means we take the first column of $\mathbf{S}$.

We recall important results on the Fast Fourier Transform [8], Fast Walsh-Hadamard Transform [14] and Parseval's theorem. Given an M-dimensional vector $\mathbf{E} = (E_1, \ldots, E_M)$ and a matrix $\mathbf{F}^{M \times M}$, we have M-dimensional vector

$$\mathbf{D} = \mathbf{FE}^T,$$

where $\mathbf{E}^T$ is the transpose of $\mathbf{E}$ and $\mathbf{F}$ is a Hadamard matrix if $\mathbf{F}(i,j) = (-1)^{ij}$, for all $i, j = 0, \cdots, M-1$. If matrix $\mathbf{F}$ is either Fourier or Hadamard, vector $\mathbf{D}$ can be computed with complexity $\mathcal{O}(M \log M)$ instead of $\mathcal{O}(M^2)$ by Fast Fourier Transform or Fast Walsh-Hadamard Transform, respectively. We recall basic facts due to Parseval:

Let $f : V_m \to \mathcal{R}$ where $\mathcal{R}$ is the real field, and $a \in V_m$. We define $f_1(a) = \sum_{b \in V_m} (-1)^{ab} f(b)$, $A = \sum_{a \in V_m} f^2(a)$ and $A_1 = \sum_{a \in V_m} f_1^2(a)$. Then

$$2^m A = A_1$$

or

$$2^m \Big(\sum_{a \in V_m} f^2(a) \Big) = \sum_{a \in V_m} f_1^2(a). \tag{2}$$

3 Statistical Model and Algorithm of Hermelin *et al.*

3.1 Constructing Multidimensional Probability Distribution

Let $f : V_l \to V_n$ be a vectorial Boolean function and binary vectors $w_i \in V_n, u_i \in V_l, i = 1, \ldots, m$ be selection patterns such that pairs of input and output masks (u_i, w_i) are linearly independent. Define the functions g_i as

$$g_i(\eta) = w_i f(\eta) \oplus u_i \eta, \quad \forall \eta \in V_l, i = 1, \cdots, m$$

and g_i has correlation ρ_i, $i = 1, \cdots, m$. Then $\rho_1, \ldots, \rho_m$ are called the base-correlations, and $g_1, \ldots, g_m$ are the base approximations of f. Let $g = (g_1, \ldots, g_m)$ be an m-dimensional vectorial Boolean function, and matrices $W = (w_1, \ldots, w_m)$ and $U = (u_1, \ldots, u_m)$ contain the output and input masks for each of the g_i, then we find the p.d. $p = (p_0, \ldots, p_{2^m - 1})$ of

$$g(\eta) = W f(\eta) \oplus U \eta.$$

Lemma 1. *[10] Let $g = (g_1, \ldots, g_m) : V_l \to V_m$ be a vectorial Boolean function and $p = (p_0, \ldots, p_{2^m-1})$ its p.d. Then*

$$2^l p_\eta = 2^{-m} \sum_{a \in V_m} \sum_{b \in V_l} (-1)^{a(g(b) \oplus \eta)}.$$

Define

$$\rho(a) = 2^{-l} \sum_{b \in V_l} (-1)^{ag(b)} = Pr(ag(X) = 0) - Pr(ag(X) = 1),$$

where X is an r.v. uniformly distributed in V_l.

Corollary 1. *Let $g : V_n \to V_m$ be a Boolean function with p.d. p and correlations $\rho(a)$ of the combined approximations ag, for all $a \in V_m$. Then for $\eta \in V_m$,*

$$p_\eta = 2^{-m} \sum_{a \in V_m} (-1)^{a\eta} \rho(a). \tag{3}$$

3.2 Multidimensional Generalization of Matsui's Alg. 1

We describe the core idea of the multidimensional algorithm 1 in [10]. Let there be m linear approximations $g_i := u_iX \oplus v_iY \oplus c_iK$, $(i = 1, \cdots, m)$. Their corresponding correlations are

$$\rho_i := 2Pr(u_iX \oplus v_iY \oplus c_iK = 0) - 1$$

where the masks c_i for the extended key K are linearly independent. In addition, the pairs of input and output masks (u_i, v_i) are linearly independent.

Define $g := (g_1, \ldots, g_m)$ with p.d. p, and $h := (h_1, \ldots, h_m)$, where $h_i = u_iX \oplus v_iY$. We call h an experimental function, as we use two of its probability distributions, namely, the theoretical p.d. q and the empirical p.d. $\hat{q}$. In the attack, q is approximated by $\hat{q}$, which is computed from N samples. Let $w_i = c_iK$, $i = 1, \cdots, m$ be the parity bits of the extended key K. As the c_i's are linearly independent, $w = (w_1, \ldots, w_m)$ defines a key class of K. Thus we have

$$g = h \oplus w.$$

Hence, the p.d. q of experimental function h is a permutation of p. Since $\{c_i\}$ are linearly independent, with the 2^m possible parity vectors w, the key space K can be classified into 2^m classes. Let w^* be the correct key class, as p^{w^*} is the permutation of p corresponding to w^*, we have $q = p^{w^*}$. By [3], given w, the relationship between p^w and p is

$$p^w_\eta = \sum_{a \in V_m} (-1)^{a(\eta \oplus w)} \rho(a) = p_{\eta \oplus w}, \quad \forall \eta \in V_m. \tag{4}$$

The KL distance is then used to determine the correct key class w^*, by constructing a hypothesis testing problem of finding the closest distribution p^w with $\hat{q}$ among the 2^m possibilities of w. In [10] the following theorem is described.

Theorem 3. *Let us have an $|\mathcal{Z}|$-ary hypothesis problem, with $|\mathcal{Z}|$ hypotheses H_w stating that the data originates from p^w, where $w \in \mathcal{Z}$ corresponds to the key. The hypothesis for which the Kullblack-Leibler distance $D(\hat{q}||p^w)$ is smallest is selected. Given some success probability P_{sc}, the lower bound N for the amount of data required to give the smallest value of the statistic when the correct key is used, is given by*

$$N \approx \frac{4 \log_2 |\mathcal{Z}|}{min_{w \neq 0} C(p^0, p^w)}. \tag{5}$$

Now we analyze the multidimensional algorithm 1 step by step.

Algorithm of Hermelin *et al.* [10]

Input of offline stage: m linear approximations $g_i, i = 1, \cdots, m$ with correlation ρ_i and p, the p.d. of g.

Offline: Compute N based on (5) and p.

Input of online stage: N pairs of (X, Y), with N computed in offline stage.

Online:

1 Distillation phase: Compute empirical p.d. $\hat{q}$ of h using 2^m counters.

2 Analysis phase:

– Construct matrix $\mathbf{T}^{2^m \times 2^m}$, with cell $\mathbf{T}(w, \eta) = \log(\frac{\hat{q}_\eta}{p^w_\eta})$, for all $w, \eta \in V_m$.

– Compute $D(\hat{q}||p^w)$, for all $w \in V_m$,

$$D = \mathbf{T}\hat{q}^T = (D(\hat{q}||p^0), \ldots, D(\hat{q}||p^{2^m-1})). \tag{6}$$

3 Sorting phase: Sort the list of w with $D(\hat{q}||p^w)$ in ascending order.

4 Searching phase: Choose the correct key class as the first element in the sorted list.

Note: in Step 2 the matrix $\mathbf{T}$ does not need to be stored. Storing a single row is sufficient. Other rows can be obtained as (4) by permuting this row when it's required for computation.

4 The Improvements

4.1 A Formula for N, the Number of Samples Required

Given $g = (g_1, \ldots, g_m)$ and $\rho(a)$, for all $a \in V_m$, by (5), the capacity of the attack of [10] is $\min_{w \neq 0} C(p^0||p^w)$. Let $p = p^0$ and $q = p^w$, we derive the following lemma. The proof is given in Appendix A.

Lemma 2.

$$C(p||q) \geq 2 \sum_{\forall a \in V_m \setminus \{0\}} \rho^2(a).$$

Combining with (5) we have the following theorem.

Theorem 4. *The estimation for N in the attack of [10] with m linear approximations g_i, $(i = 1, \ldots, m)$, where the $\{c_i\}$ are linearly independent and (u_i, v_i) are linearly independent, is given by*

$$N \approx \frac{m}{2 \sum_{\forall a \in V_m \setminus \{0\}} \epsilon^2(a)}. \tag{7}$$

With the new formula for N, we can now compare the multidimensional Alg. 1 with previous attacks on their data complexities, as shown in Table 1. The last row contains our formula for N. We can see that the framework provided by Hermelin *et al.* in [10] is able to exploit all the combined approximations systematically, as compared to Biryukov *et al.* in [2].

Table 1. Comparisons of data complexities with different attack frameworks

N	Framework	Comments
$1/\epsilon^2$	[12] Matsui	
$1/\sum_{i=1}^{m} \epsilon_i^2$	[11] Kaliski and Robshaw	
$1/\sum_{i=1}^{m} \epsilon_i^2$	[2] Biryukov *et al.*	
$m/2 \sum_{a \neq 0} \epsilon^2(a)$	[10] Hermelin *et al.*	See (7)

4.2 Analysis of the Multidimensional Alg. 1 in [10]

We analyze the multidimensional Alg. 1 of Hermelin *et al.* step by step and introduce our improvements.

Identifying the Bottleneck – Complexity Analysis. Offline: An estimation of N can be computed from (5), which is slow – to the best of our knowledge, no obvious algorithm is much faster than $3m2^m$ steps.

Online:

1 Distillation Phase: Using 2^m counters to compute $\hat{q}$ from N samples. The complexity is $2^m N$.

2 Analysis Phase: Computing D has a time complexity of $O(2^{2m})$.

If we increase m, the number of base approximations $g_1, \ldots, g_m$, we may expect N to decrease [10], but the complexities in the distillation and the analysis phases suffer from exponential increase. The actual number of approximations that can be used is hence limited by the computation resources allowed. It is a trade-off.

Improving the Bottleneck. To compute the KL distance between a p.d. $\hat{q}$ and p^w, we observe that after expanding $D(\hat{q}||p^w) = \sum_{\eta \in V_m} \hat{q}_\eta \log(\hat{q}_\eta / p_\eta^w) = \sum_{\eta \in V_m} \hat{q}_\eta \log \hat{q}_\eta - \sum_{\eta \in V_m} \hat{q}_\eta \log p_\eta^w$, the term $\sum_{\eta \in V_m} \hat{q}_\eta \log \hat{q}_\eta$ is a constant and hence does not affect the ranking of key classes. We can define $\bar{D}(\hat{q}||p^w) = \sum_{\eta \in V_m} \hat{q}_\eta \log p_\eta^w$ and use $\bar{D}(\cdot||\cdot)$ to rank the key class candidates. The list of w in the sorting phase of the online stage is now sorted by the values of $\bar{D}(\hat{q}||p^w)$ in descending order.

A new matrix $\bar{\mathbf{T}}$ is constructed as

$$\bar{\mathbf{T}}(w, \eta) = \log(p_\eta^w), \quad \forall w, \eta \in V_m. \tag{8}$$

In Section 4.3 we show that matrix $\bar{\mathbf{T}}$ is a circulant matrix. By Theorem 1 the Fast Fourier Transform algorithm can be applied for fast computation. For $\bar{\mathbf{T}}$, only the first column ($\bar{\mathbf{T}}(w, 0) = \log(p_0^w) = \log(p_w)$, for all $w \in V_m$) needs to be stored. The memory requirement is 2^m. $\bar{\mathbf{T}}$ is used for computing $\bar{D} = \bar{\mathbf{T}}\hat{q}^T$.

Our Improvements:

Offline: We present a formula for N based on p, without using (5).

Online:

Step 1: We present Method-A to calculate $\hat{q}$ from N samples given. The complexity is $2m2^m + mN$.

Step 2: We present Method-B to compute $\bar{D}$. The complexity is $3m2^m$.

4.3 Fast Computation of Empirical Distribution – Method A

Method A is for computing the empirical distribution $\hat{q}$ of the experimental function $h = (h_1, \ldots, h_m)$ from N samples $(\mathbf{X}, \mathbf{Y})$. First, the correlations of the combined linear approximations bh, for all $b \in V_m$,

$$\hat{\gamma}(b) = Pr(bh(\mathbf{X}, \mathbf{Y}) = 0) - Pr(bh(\mathbf{X}, \mathbf{Y}) = 1),$$

are calculated. Then the p.d. $\hat{q}$ is computed from $\hat{\gamma}(b)$ by (3).

We show how to compute $\hat{\gamma}(b)$, for all $b = (b_1, \ldots, b_m) \in V_m$ from N samples:

$$\hat{\gamma}(b) = \hat{\gamma}(\bigoplus_{i=1}^{m} b_i h_i) = \frac{\sum_{j=1}^{N} (-1)^{\bigoplus_{i=1}^{m} b_i h_i(X_j, Y_j)}}{N}$$
$$= \sum_{a \in V_m} (-1)^{\bigoplus_{i=1}^{m} b_i a_i} \frac{T_a}{N} = \sum_{a \in V_m} (-1)^{ba} \frac{T_a}{N},$$

where $a = (a_1, \ldots, a_m)$ and $T_a = \#\{(X_j, Y_j), j = 1, \ldots, N : h_i(X_j, Y_j) = a_i\}$.

Let $S^{2^m \times 2^m}$ be the matrix defined by $S(b, a) = (-1)^{ba}$, and let $E = (\frac{T_0}{N}, \ldots, \frac{T_{2^m-1}}{N})$, and $\hat{\gamma} = (\hat{\gamma}_0, \ldots, \hat{\gamma}_{2^m-1})$, then

$$\hat{\gamma}(b) = \sum_{a \in V_m} S(b, a) E_a, \forall b \in V_m, \text{ or } \hat{\gamma} = SE^T.$$

Since $\mathbf{S}$ is a Hadamard matrix, we can apply the Fast Walsh-Hadamard Transform algorithm for computing $\hat{\gamma}(b)$ for all $b \in V_m$ with complexity $m2^m$, and the storage is $\mathcal{O}(2^m)$. The complexity for computing the counter vector $T = (T_1, \ldots, T_{2^m})$ is mN, by evaluating the m base approximations against each of the N samples.

Construct the vector $R = (R_0, \ldots, R_{2^m-1})$ with $R_b = 2^{-m}\hat{\gamma}(b)$, for all $b \in V_m$. From (3) we have

$$\hat{q}_\eta = \sum_{b \in V_m} (-1)^{\eta b} R_b, \forall \eta \in V_m, \text{ or } \hat{q} = SR^T.$$

The Fast Hadamard Transform is used again to compute $\hat{q}$. Hence, the total complexity is $mN + 2m2^m$ for computing $\hat{q}$ from N samples.

To Summarize Method A

Step 1: Construct the vector E from N samples (X, Y).
Step 2: Compute $\hat{\gamma} = SE^T$, then construct the vector R.
Step 3: Compute $\hat{q} = SR^T$.

4.4 Fast Computation of Kullback-Leibler Distance – Method B

Method-B is used to compute the vector $\bar{D}$ in (6) with modified matrix $\bar{T}$ in (8) based on the idea of circulant matrix in [7]. The proof of the following theorem can be found in Appendix B.

Theorem 5. *The matrix $\bar{T}$ is level-m circulant with type* $\underbrace{(2, 2, \ldots, 2)}_{m\text{-}times}$.

Method-B: From Theorems 1, 2, and 5 we have

$$\bar{\mathbf{T}} = \mathbf{F}^* diag(\lambda)\mathbf{F}, (\mathbf{F}^*\mathbf{F} = \mathbf{I}), \tag{9}$$

$$\lambda = \mathbf{F}\bar{\mathbf{T}}(:, 1)\sqrt{2^m}, \tag{10}$$

and

$$\bar{D} = (\bar{D}(\hat{q}||p^0), \ldots, \bar{D}(\hat{q}||p^{2^m-1})) = \bar{\mathbf{T}}\hat{q} = (\mathbf{F}^*(diag(\lambda)(\mathbf{F}\hat{q}))).$$

Applying the Fast Fourier Transform three times for $\mathbf{F}$, $diag(\lambda)$, and $\mathbf{F}^*$ we get all the values $\bar{D}(\hat{q}||p^w)$, for all $w \in V_m$. The complexity is $3m2^m$.

5 Efficiency of the Improved Algorithm

The improved algorithm works as follows:

Input of offline stage: m linear approximations $g_1, \ldots, g_m$ and p.d. p and $\rho(a)$ of g, for all $a \in V_m$.

Offline stage:

Step 1: Compute N by (7).

Step 2: Compute the eigenvalue vector λ of matrix $\bar{\mathbf{T}}$ based on (10) and p.

Input of online stage : N samples (X, Y).

Online stage :

Step 1 – Distillation: Compute empirical p.d. $\hat{q}$ of h from N by Method-A.

Step 2 – Analysis: From λ and $\hat{q}$, compute vector $\bar{D}$ by Method-B.

Step 3 – Sorting: Sort the key classes w with $\bar{D}(\hat{q}||p^w)$ in descending order.

Step 4 – Searching: Choose the correct key class as the first element in the sorted list.

The improved results are presented in Table 2 to compare with previous results on the complexities of the distillation phase and the analysis phase, following [10]. In practise mN is always larger than $2m2^m$, and we estimate $\mathcal{O}(2m2^m + mN) = \mathcal{O}(mN)$.

Following the definitions of $N_{s.i.}$ and N_{plain} in [10], $N < N_{s.i.} < N_{plain}$, $m < m'$ (also see Table 1), we have significant improvement in the time complexities

Table 2. Complexity comparison between different algorithms

	Distillation Phase				Analysis Phase			
	Plain	Biryukov	Hermelin	Method-A	Plain	Biryukov	Hermelin	Method-B
Data	$\mathcal{O}(N_{plain})$	$\mathcal{O}(N_{s.i})$	$\mathcal{O}(N)$	$\mathcal{O}(N)$	-	-	-	-
Time	$\mathcal{O}(mN_{plain})$	$\mathcal{O}(m' N_{s.i})$	$\mathcal{O}(2^m N)$	$\mathcal{O}(mN)$	$\mathcal{O}(m2^m)$	$\mathcal{O}(m'2^m)$	$\mathcal{O}(2^{2m})$	$\mathcal{O}(m2^m)$
Mem	$\mathcal{O}(m)$	$\mathcal{O}(m')$	$\mathcal{O}(2^m)$	$\mathcal{O}(2^m)$	$\mathcal{O}(2^m)$	$\mathcal{O}(2^m)$	$\mathcal{O}(2^m)$	$\mathcal{O}(2^m)$

for the distillation phase and the analysis phase. In addition, we have proved that the multidimensional algorithm 1 requires fewer samples than Biryukov *et al.*[2]. The same result was observed in [10] with only empirical evidence.

6 Application to Cryptanalysis

In this section we apply the improved algorithm to reduced-round Serpent and derive the attack complexities. We derive attack scenarios by using 4-round and 9-round linear characteristics used in [10] and [4] to show that the improvement made in this paper can improve previous cryptanalysis by many orders of magnitude.

6.1 Application to the 4-Round Serpent Scenario

In [7] 64 approximations were used to obtain 10 parity bits of 4-round Serpent, from S_4 to S_7. The approximations are modified from the first 4 rounds of the 6-round linear characteristic of [4], with the details described in [5]. As shown in [10], these approximations used are not linearly independent in text masks and key masks. There are 8 of them with correlation in magnitude of 2^{-11} and 56 of 2^{-12}. This gives an overall capacity of $4\sum_{i=0}^{63}\epsilon_i^2 = 2^{-17.54}$ hence estimation of $4m/4\sum_{i=1}^{m'}\epsilon_i^2 = 2^{22.86}$ for N. By selecting a basis of 10 approximations $L_0, \ldots, L_9$ from the 64, where L_i is $u_iX \oplus w_0Y \oplus c_iK = 0$, Hermelin *et al.* [10] studied all the approximations generated as in $span\{L_0, \ldots, L_9\}$. Of the 1023 combinations, 8, 64 and 128 are with non-negligible correlation in magnitude of 2^{-11}, 2^{-12} and 2^{-13}, respectively. Lemma 2 gives capacity $C(p||q)$ of at least $2\sum_{a\neq 0}\rho^2(a) = 2\cdot(8\cdot 2^{-22} + 64\cdot 2^{-24} + 128\cdot 2^{-26}) = 2^{-16}$ and hence the estimation for N is $4m/C(p||q) = 2^{21.3}$, in perfect correspondence to the experimental results in [10]. With Method-A and Method-B, we can set $m = 16$, yielding an attack with better complexity than the multidimensional Alg. 1 of [10] with $m = 10$.

The approximations $L_0, \ldots, L_9$ are derived from the same linear characteristic with input masks $u_0, \ldots, u_9$ and the same output mask w_0. We obtain 6 additional ciphertext masks $w_1, \ldots, w_6$ and use the following 16 approximations as base approximations: $(u_0, w_0), \ldots, (u_9, w_0), (u_1, w_1), \ldots, (u_1, w_6)$. An exhaustive check of all combined correlations shows that 32, 384, 1664, 3072 and 2048 of the combined approximations have correlation in magnitude of 2^{-11}, 2^{-12}, 2^{-13}, 2^{-14} and 2^{-15} respectively. This gives a capacity of $2^{-12.8}$, hence we estimate $N \sim 4m/C(p||q) = 2^{18.8}$. We tabulate the comparisons in Table 3. From the table we can conclude that it is clearly advantageous to be able to have a larger m, i.e., when using $m = 16$ instead of $m = 10$, in the case of Serpent. A larger m in this case makes it possible to have a larger number of non-negligible approximations, hence larger capacity, which implies reduced data complexity thus the overall time complexity. Setting $m = 16$ improves the cryptanalysis due to the fact that the number of non-negligible approximations in Serpent is exponential in m. It may appear that the attack is slower with a larger m. However, in fact, it

Table 3. Attack complexities on 4-round Serpent

m	$C(p\|\|q)$	N	Distillation Phase		Analysis Phase		Memory
			Hermelin *et al.* [10]	This paper	Hermelin *et al.* [10]	This paper	
10	2^{-16}	$2^{21.3}$	$2^{31.3}$	$2^{24.6}$	2^{20}	$2^{14.9}$	2^{10}
16	$2^{-12.8}$	$2^{18.8}$	$2^{34.8}$	$2^{23.2}$	2^{32}	$2^{21.6}$	2^{16}

is the reduction in data complexity N that dominates the time complexity, so we obtain a faster attack. Essentially, this is a trade-off and it is always meaningful to find an appropriate m for a block cipher to produce optimal attack complexity. However, by our experiments, in the case of DES, the number of high probability approximations is much fewer than that of Serpent, so larger values of m do not give better results than fewer approximations. We believe that SPN block ciphers with small S-boxes are more likely to be vulnerable to our attack.

6.2 Multidimensional Linear Cryptanalysis of 9-Round Serpent

We take the 9-round linear characteristic of [4], where the details are described in [5]. This linear characteristic starts from S_3 and ends after the next S_3, with correlation 2^{-49}. The first round has 11 active S-boxes, which results in a correlation of 2^{-11} and the remaining 8 rounds with correlation 2^{-38}. By modification to the input masks, there is a total of 10^{11} masks, with the magnitude of first round correlation from 2^{-11} to 2^{-22}. By picking 44 independent base input masks, we can expect to have around 10^{11} out of the 2^{44} combined approximations giving non-negligible correlations. We can exploit the huge number of approximations by a 44-dimensional attack, with capacity

$$C(p||q) = 2 \cdot [\sum_{i=0}^{11} \binom{11}{i} \cdot 2^i \cdot 8^{11-i} \cdot ((2^{-1})^i \cdot (2^{-2})^{11-i})^2] \cdot (2^{-38})^2 = 2^{-75}$$

which is 2^{22} times larger than the capacity 2^{-98} for the single approximation scenario. Correspondingly the estimation for N is $4m/C(p||q) = 2^{82.5}$. The time complexity for the distillation phase is 2^{88} and the analysis phase is 2^{51}. Around 2^{44} memory are needed. In [4], this 9-round linear characteristic is used to break a 10-round and an 11-round Serpent with Matsui's Alg. 2 extended with multiple approximations. It requires at least 2^{99} known-plaintext which is much higher than our estimation. Moreover, the time complexities presented in [4] did not take into account the distillation phase, which should require no lower than mN which is greater than 2^{99}. Hence, it is much higher than our overall time complexity 2^{88}.

Comparing with extensions of Alg. 2, a disadvantage of Alg. 1 is that an r-round linear characteristic can be used to attack an r-round block cipher. However, it is noted in [7] that optimal application of Algorithm 2 with multiple approximations requires accurate estimation of the biases, which can be unreliable if multiple linear characteristics exist with non-negligible probability under

the same text mask. Intuitively, Alg. 1 is likely to give more reliable estimation for theoretical cryptanalysis.

7 Conclusions

In this paper, we have presented a new formula for N, in terms of $\rho(a)$ and m, for the $|\mathcal{Z}|$-ary hypothesis testing problem in the multidimensional generalization of Matsui's Algorithm 1 in [10]. The number of known plaintext needed can now be computed from $\rho(a)$ directly, whereas it is harder to compute $\min_{w \neq 0} C(p^0, p^w)$ in the original formula (5).

Method-A and Method-B have been presented to compute the empirical distribution and Kullback-Leibler distance, as improvements to the multidimensional Alg. 1 in [10]. A significant reduction of time complexity in the distillation and analysis phases can be achieved. Breaking these bottlenecks allows many more base approximations to be used.

As shown in the case of 4- and 9-round Serpent, the increase in m brings significant reduction on the data complexity and the time complexity. We expect this improved algorithm to outperform previous multiple linear cryptanalysis. We observed that the framework of Hermelin *et al.* [10], with our improvement, solves the problem of Biryukov *et al.* [2] that the statistical independence assumption cannot in general be guaranteed in practice, and in fact, does not need to be guaranteed, because there exists a large number of combined approximations with non-negligible correlations. Meanwhile, the series of experiments in [4] and [7] provide excellent examples for this argument.

It also implies that, for block cipher designs, bounding the maximum correlation for any single linear characteristic is not sufficent to claim security. Especially for SPN block ciphers with small S-boxes, a single linear trail with multiple active S-boxes in the first or last round can be modified to have many approximations, exponential in the number of active S-boxes of outer rounds. When these approximations are with similar magnitude of correlations, multidimensional linear cryptanalysis as a systematic way to exploit these combined correlations can reduce the attack complexity greatly. It's worthy for designers to have larger security margins or to try to develop specific mechanisms to prevent an attacker from forming an exponential number of valid linear approximations.

Acknowledgements. We thank Joo Yeon Cho for providing the linear approximations used in [10]. This reserach is supported by the Singapore National Research Foundation under Research Grant NRF-CRP2-2007-03 and the Singapore Ministry of Education under Research Grant T206B2204. The first author is supported by the Singapore International Graduate (SINGA) Scholarship.

References

1. 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)

2. Biryukov, A., De Cannière, C., Quisquater, M.: On Multiple Linear Approximations. In: Franklin, M. K. (ed.) CRYPTO 2004. LNCS, vol. 3152, pp. 1–22. Springer, Heidelberg (2004)
3. Cho, J.Y., Hermelin, M., Nyberg, K.: A New Technique for Multidimensional Linear Cryptanalysis with Applications On Reduced Round Serpent. In: Lee, P.J., Cheon, J.H. (eds.) ICISC 2008. LNCS, vol. 5461, pp. 383–398. Springer, Heidelberg (2009)
4. Collard, B., Standaert, F.-X., Quisquater, J.-J.: Improved and Multiple Linear Cryptanalysis of Reduced Round Serpent. In: Pei, D., Yung, M., Lin, D., Wu, C. (eds.) Inscrypt 2007. LNCS, vol. 4990, pp. 51–65. Springer, Heidelberg (2008)
5. Collard, B., Standaert, F.-X., Quisquater, J.-J.: Improved and Multiple Linear Cryptanalysis of Reduced Round Serpent - Description of the Linear Approximations, 2007 (unpublished manuscript)
6. Collard, B., Standaert, F.-X., Quisquater, J.-J.: Improving the Time Complexity of Matsui's Linear Cryptanalysis. In: Nam, K.-H., Rhee, G. (eds.) ICISC 2007. LNCS, vol. 4817, pp. 77–88. Springer, Heidelberg (2007)
7. Collard, B., Standaert, F.-X., Quisquater, J.-J.: Experiments on the Multiple Linear Cryptanalysis of Reduced Round Serpent. In: Nyberg, K. (ed.) FSE 2008. LNCS, vol. 5086, pp. 382–397. Springer, Heidelberg (2008)
8. Cormen, T.H., Stein, C., Rivest, R.L., Leiserson, C.E.: Introduction to Algorithms. McGraw-Hill Higher Education, New York (2001)
9. Desmedt, Y.G. (ed.): CRYPTO 1994. LNCS, vol. 839. Springer, Heidelberg (1994)
10. Hermelin, M., Cho, J.Y., Nyberg, K.: Multidimensional Linear Cryptanalysis of Reduced Round Serpent. In: Mu, Y., Susilo, W., Seberry, J. (eds.) ACISP 2008. LNCS, vol. 5107, pp. 203–215. Springer, Heidelberg (2008)
11. Kaliski Jr., B.S., Robshaw, M.J.B.: Linear Cryptanalysis Using Multiple Approximations. In: Desmedt (ed.) [9], pp. 26–39
12. Matsui, M.: Linear Cryptanalysis Method for DES Cipher. In: Helleseth, T. (ed.) EUROCRYPT 1993. LNCS, vol. 765, pp. 386–397. Springer, Heidelberg (1994)
13. Matsui, M.: The First Experimental Cryptanalysis of the Data Encryption Standard. In: Desmedt [9], pp. 1–11
14. Rao Yarlagadda, R.K., Hershey, J.E.: Hadamard Matrix Analysis and Synthesis: with Applications to Communications and Signal/image Processing. Kluwer Academic Publishers, Norwell (1997)

Appendix

A Proof for Lemma 2

Proof. Let $g = (g_1, \ldots, g_m)$ and given $\rho(a)$, for all $a \in V_m$, from (2), (3)

$$2^m(\sum p_\eta^2) = \sum \rho^2(a).$$

We have 2 facts: $p^0 = p$ and p^w is a permutation of p. From (1), we replace p^0 and p^w by p, q respectively in (5). Define $u = \max\{q_\eta, \eta \in V_m\} = \max\{p_\eta, \eta \in V_m\}$. Then

$$C(p||q) = \sum_{\eta \in V_m} \frac{(p_\eta - q_\eta)^2}{q_\eta} \geq \frac{1}{u} \sum_{\eta \in V_m} (p_\eta^2 + q_\eta^2 - 2p_\eta q_\eta).$$

We have

$$\frac{q_\eta}{u} \leq 1 \Rightarrow -\frac{q_\eta}{u} \geq -1.$$

Since q is a permutation of p:

$$\sum_{\eta \in V_m} p_\eta = 1 \text{ and } \sum_{\eta \in V_m} p_\eta^2 = \sum_{\eta \in V_m} q_\eta^2.$$

Hence

$$\begin{aligned}\frac{1}{u}\sum_{\eta \in V_m}(p_\eta^2 + q_\eta^2 - 2p_\eta q_\eta) &= \frac{2}{u}(\sum_{\eta \in V_m} p_\eta^2) - 2\sum_{\eta \in V_m}\frac{q_\eta}{u}p_\eta \\ &\geq \frac{2 \cdot 2^{-m}\sum_{a \in V_m}\rho^2(a)}{u} - 2(\sum_{\eta \in V_m} p_\eta) \\ &= \frac{2 \cdot 2^{-m}\sum_{a \in V_m}\rho^2(a)}{u} - 2.\end{aligned}$$

From (3),

$$u = \max\{p_\eta\} \leq 2^{-m}(\sum_{a \in V_m}|\rho(a)|),$$

so that

$$C(p||q) \geq \frac{2\sum_{a \in V_m}\rho^2(a)}{\sum_{a \in V_m}|\rho(a)|} - 2.$$

In practice, since $\rho(0) = 1$ and $\rho(a) \ll 1,\ \forall a \neq 0$, we have $\sum_{a \in V_m}|\rho(a)| \approx 1$. Hence

$$C(p||q) \geq 2\sum_{\forall a \in V_m \setminus \{0\}}\rho^2(a).$$

B Proof for Theorem 5

Proof. The structure of the modified matrix $\bar{T}^{2^m \times 2^m}$ is

$$\begin{pmatrix} \log(p_0^0) & \log(p_1^0) & \cdots & \log(p_{2^m-1}^0) \\ \log(p_0^1) & \log(p_0^1) & \cdots & \log(p_{2^m-1}^1) \\ \vdots & \vdots & \ddots & \vdots \\ \log(p_0^{2^m-1}) & \log(p_1^{2^m-1}) & \cdots & \log(p_{2^m-1}^{2^m-1}) \end{pmatrix}.$$

From the relation between p^w $(\forall w \neq 0)$ and $p^0 = p$:

$$p_i^w = \sum_{a \in V_m}(-1)^{a(w \oplus i)}\rho(a) = \sum_{a \in V_m}(-1)^{aj}\rho(a) = p_j,$$

where $j = w \oplus i$.

Hence, $\log(p_i^w) = \log(p_j), j = w \oplus i$. It means that $\bar{T}(w,i) = \bar{T}(0,j) = \bar{T}(0, w \oplus i)$.

Divide $\bar{T}$ into 4 blocks, each with size $(2^{m-1} \times 2^{m-1})$:

$$\begin{pmatrix} \bar{T}_{11} & \bar{T}_{12} \\ \bar{T}_{21} & \bar{T}_{22} \end{pmatrix}.$$

Then for $0 \leq i, j \leq 2^{m-1} - 1$:

- $\bar{T}_{11}(i,j) = \bar{T}(i,j) = \bar{T}(0, i \oplus j)$,
- $\bar{T}_{21}(i,j) = \bar{T}(i + 2^{m-1}, j) = \bar{T}(0, (i + 2^{m-1}) \oplus j) = \bar{T}(0, i \oplus j \oplus 2^{m-1})$,
- $\bar{T}_{12}(i,j) = \bar{T}(i, j + 2^{m-1}) = \bar{T}(0, i \oplus (j + 2^{m-1})) = \bar{T}(0, (i + 2^{m-1}) \oplus j)$,
- $\bar{T}_{22}(i,j) = \bar{T}(i+2^{m-1}, j+2^{m-1}) = \bar{T}(0, (i+2^{m-1}) \oplus (j+2^{m-1})) = \bar{T}(0, i \oplus j)$.

Consequently, $\bar{T}_{11} = \bar{T}_{22}$ and $\bar{T}_{12} = \bar{T}_{21}$, hence $\bar{T}$ is 2-block circulant. We can inductively repeat the same argument to $\bar{T}_{11}$ with $m = m - 1$. Since $\bar{T}_{12} = \bar{T}_{11} \oplus 2^{m-1}$, the structure of $\bar{T}_{12}$ is similar to the circulant structure of $\bar{T}_{11}$. Hence, the matrix $\bar{T}$ is level-m circulant with type $\underbrace{(2, 2, \ldots, 2)}_{\text{m-times}}$.