Optimal Key Ranking Procedures in a Statistical Cryptanalysis

Pascal Junod and Serge Vaudenay

Security and Cryptography Laboratory
Swiss Federal Institute of Technology
CH-1015 Lausanne, Switzerland
{pascal.junod, serge.vaudenay}@epfl.ch

Abstract. Hypothesis tests have been used in the past as a tool in a cryptanalytic context. In this paper, we propose to use this paradigm and define a precise and sound statistical framework in order to optimally mix information on independent attacked subkey bits obtained from any kind of statistical cryptanalysis. In the context of linear cryptanalysis, we prove that the best mixing paradigm consists of sorting key candidates by decreasing weighted Euclidean norm of the bias vector.

Keywords: Key ranking, statistical cryptanalysis, Neyman-Pearson lemma, linear cryptanalysis

1 Introduction

Historically, statistical hypothesis tests, although well-known in many engineering fields, has not been an explicitely widely-used tool in the cryptanalysis of block ciphers. Often, some distinguishing procedures between two statistical distributions are proposed, but without much attention on their optimality. To the best of our knowledge, an unpublished report of Murphy, Piper, Walker and Wild [MPWW95] is the first work where the concept of statistical hypothesis tests is discussed in the context of "modern" cryptanalysis. More recently, a paper of Fluhrer and McGrew [FM01] discussed the performances of an optimal statistical distinguisher in the cryptanalysis of a stream cipher. These tools were again used by Mironov [Mir02], by Coppersmith *et al.* [CHJ02], by Golić and Menicocci [GM] in the same context, for instance, while Junod [Jun03] makes use of them for deriving the asymptotic behaviour of some optimal distinguishers.

1.1 Contributions of this Paper

In this paper, we propose a sound and precise statistical cryptanalytic framework which extends Vaudenay's one [Vau96]; furthermore, we describe an *optimal* distinguishing procedure that can be employed during any statistical cryptanalysis involving subkey candidates ranking.

As illustration, we apply this distinguishing procedure to the linear cryptanalysis of DES [DES77] as proposed by Matsui in [Mat94]. In the first version of linear cryptanalysis of DES [Mat93], Matsui's attack returns a subkey which is the correct one with high probability, while a refined version of the attack [Mat94] returns a list of subkeys sorted by *maximum-likelihood*. This approach, which is very similar to the list decoding paradigm in coding theory, allows to decrease the number of known plaintext-ciphertext pairs needed. Although very simple to implement, Matsui's key ranking heuristic is however not optimal. We show basically that by sorting the subkey candidates by decreasing *sum of squares of the experimental biases*, we obtain a ranking procedure which minimizes the costs of the attack's exhaustive search part.

At first sight, the optimisation of the exhaustive search complexity of a cryptanalysis does not seem to be so interesting, since exhaustive search is a "cheap" operation for a cryptanalyst, compared to the cost, or the difficulty of finding the required amount of known plaintext-ciphertext pairs. However, we show in this paper that by optimising the exhaustive search part of a linear cryptanalysis of DES, it is possible to decrease in a sensible way the number of pairs needed and to keep the computational complexity within a reasonable area. In [Jun01], Junod did a complexity analysis and proved that Matsui's attack against DES performs better than expected, which was already conjectured. He further confirmed this fact experimentally with 21 linear cryptanalysis: given 2^{43} known plaintext-ciphertext pairs, and a success probability equal to 85 %, the computational complexity had an upper bound of $2^{40.75}$ DES evaluations. In this paper, the power of this technique is illustrated by experimentally demonstrating that one can decrease the computational complexity of Matsui's attack against DES by an average factor of two, or, equivalently, decrease the number of known plaintext-ciphertext pairs needed by a non-trivial factor (*i.e* 31 %) without an explosion of the computational complexity (*i.e.* less than 2^{45} DES evaluations); one can also *divide the number of known pairs by two* (*i.e.* to 2^{42}) while keeping the computational complexity within 2^{47} DES evaluations.

Other examples of potential *direct* application of our optimal ranking rule are Shimoyama-Kaneko's attack [SK98] on DES which uses quadratic boolean relations, or Knudsen and Mathiassen's *chosen*-plaintexts version [KM01] of linear cryptanalysis against DES. However, the ideas behind our ranking method are not restricted to those attacks and may be applied in any statistical cryptanalysis.

The rest of this paper is organized as follows: in §2, we recall Vaudenay's statistical cryptanalysis model and Matsui's ranking procedures; in §3, we introduce the necessary statistical tools and we propose the *Neyman-Pearson ranking procedure*. In §4, we apply it to a linear cryptanalysis of DES, we present some experimental results on the improvment and we discuss potential applications to other known attacks. Finally, we give some concluding remarks in §5.

1.2 Notation

The following notation will be used throughout this paper. Random variables $X, Y, \ldots$ are denoted by capital letters, while *realizations* $x \in \mathcal{X}, y \in \mathcal{Y}, \ldots$ of random variables are denoted by small letters. The fact for a random variable X to follow a distribution D is denoted $X \leftarrow \mathsf{D}$, while its probability density and distribution functions are denoted by $f_{\mathsf{D}}(x)$ and $F_{\mathsf{D}}(x) = \Pr_{X \leftarrow \mathsf{D}}[X \leq x] = \int_{-\infty}^{x} f_{\mathsf{D}}(t)dt$, respectively. When the context is clear, we will write simply $\Pr[X \leq x]$. Finally, as usual, "iid" means "independent and identically distributed".

2 Statistical Cryptanalysis and Key Ranking Procedures

In this paper, we will assume that a given cryptanalysis can be seen as a *statistical cryptanalysis*, in the sense of Vaudenay's model [Vau96], and that it uses a key ranking procedure.

2.1 Statistical Cryptanalysis

We recall now briefly the principles of a statistical cryptanalysis. Let $\mathcal{P}$, $\mathcal{C}$ and $\mathcal{K}$ be the plaintext, cipertext and key space, respectively. A statistical cryptanalysis uses three functions, denoted f_1, f_2 and f_3 which have the following role:

- $f_1 : \mathcal{K} \rightarrow \mathcal{L}$ is a function which eliminates information of the key unrelated to the cryptanalysis.
- $f_2 : \mathcal{P} \times \mathcal{C} \rightarrow \mathcal{S}$, where $\mathcal{S}$ is called the *sample space*, eliminates information about the plaintext and ciphertext spaces unrelated to the attack.
- $f_3 : \mathcal{L} \times \mathcal{S} \rightarrow \mathcal{Q}$, where $\mathcal{Q}$ is a space summarizing information depending on intermediate results in the encryption.

1. **Counting Phase**: Collect several random samples $s_j = f_2(P_j, C_j)$, for $j = 1, \ldots, n$ and count all occurences of all the possible values of the s_j's in $|\mathcal{S}|$ counters.
2. **Analysis Phase**: For each of the subkey candidates ℓ_i, $1 \leq i \leq |\mathcal{L}|$, count all the occurences in all $x_i = h_3(\ell_i, s_j)$ and give it a mark μ_{ℓ_i} using the statistic $\Sigma(x_1, \ldots, x_n)$.
3. **Sorting Phase**: Sort all the candidates ℓ_i using their mark μ_{ℓ_i}. This list of sorted candidates is denoted $\mathcal{U}$.
4. **Searching Phase**: Exhaustively try all keys following the sorted list of all the subkey candidates.

Fig. 1. Structure of a statistical cryptanalysis

In order to be efficient, a statistical cryptanalysis should fulfil the following conditions: the information $x = f_3(\ell, s)$, where $\ell \in \mathcal{L}$, $s \in \mathcal{S}$ and $x \in \mathcal{Q}$, should be computable with *small* pieces of information on $(p, c) \in \mathcal{P} \times \mathcal{C}$ and $k \in \mathcal{K}$ (namely, s and ℓ); furthermore, the information $x = f_3(s, \ell_r)$ should be statistically distinguishable from $x' = f_3(s, \ell_w)$, where ℓ_r and ℓ_w is the information given by the right key and a wrong key, respectively. The main idea of the attack consists in assuming that we can distinguish the right key from wrong key with help of a statistical measurement Σ on the observed distribution of the x_i's. The attack is described in Fig. 1. The *data complexity* is then defined to be the number n of known plaintext-ciphertext pairs needed in step 1, while the *computational complexity* is defined to be the number operations in the last phase of the attack. We note that usually, the complexity of steps 2 and 3 is negligible, but it may not be the case in all situations.

Key ranking is a technique introduced by Matsui in [Mat94] in order to increase the success probability of a linear cryptanalysis against DES; it corresponds to step 4 in Fig. 1: instead of returning the subkey $\ell_{\max}$ possessing the highest mark $\mu_{\ell_{\max}} \triangleq \max_i \mu_{\ell_i}$ out of $|\mathcal{L}|$ subkey candidates, the idea is to return a *sorted list* $\mathcal{U}$ containing key candidates ranked by likelihood and to search for the remaining unattacked bits in this order.

Obviously, two central points in a statistical cryptanalysis are the definition of the statistic Σ and of the mark μ which has to be assigned to a subkey candidate. The first issue is the essence of the attack: the cryptanalyst must find a "statistical weakness" in the cipher. In the section §3, we will address the second issue in a general way by using concepts of statistical hypothesis testing and we consider known techniques under this light; before, we recall some generic facts about linear cryptanalysis and the related ranking procedures proposed by Matsui.

2.2 Linear Cryptanalysis and Related Ranking Procedures

We recall briefly the principles of a linear cryptanalysis. The attack's core is unbalanced linear expressions, i.e. equations involving a modulo two sum of plaintext and ciphertext bits on the left and a modulo two sum of key bits on the right. Such an expression is unbalanced if it is satisfied with probability[1]

$$p \triangleq \frac{1}{2} + \epsilon, 0 < |\epsilon| \leq \frac{1}{2} \tag{1}$$

when the plaintexts and the key are independent and chosen uniformly at random.

Given some plaintext bits $\mathsf{P}_{i_1}, \ldots, \mathsf{P}_{i_r}$, ciphertext bits $\mathsf{C}_{j_1}, \ldots, \mathsf{C}_{j_s}$ and key bits $\mathsf{K}_{k_1}, \ldots, \mathsf{K}_{k_t}$, and using the notation $\mathsf{X}_{[l_1,\ldots,l_u]} \triangleq \mathsf{X}_{l_1} \oplus \mathsf{X}_{l_2} \oplus \ldots \oplus \mathsf{X}_{l_u}$, we can write a linear expression as

$$\mathsf{P}_{[i_1,\ldots,i_r]} \oplus \mathsf{C}_{[j_1,\ldots,j_s]} = \mathsf{K}_{[k_1,\ldots,k_t]} \tag{2}$$

As this equation allows to get only one bit of information about the key, one usually use a linear expression spanning all the rounds but one; it is possible to identify the subkey involved in the last round. One can rewrite (2) as

$$\mathsf{P}_{[i_1,\ldots,i_r]} \oplus \mathsf{C}_{[j_1,\ldots,j_s]} \oplus F^{(r)}_{[m_1,\ldots,m_v]}\left(\mathsf{C}, \mathsf{K}^{(r)}\right) = \mathsf{K}_{[k_1,\ldots,k_t]} \tag{3}$$

Now, one can easily identify the abstract spaces defined in the generic model of Fig. 1: the (sub)key space $\mathcal{L}$ is the set of all possible values of the involved subkey (*i.e.* the "interesting" bits of $\mathsf{K}^{(r)}$ and those of $\mathsf{K}_{[k_1,\ldots,k_t]}$); the sample space $\mathcal{S}$ is the set of all possible $\mathsf{P}_{i_1}, \ldots, \mathsf{P}_{i_r}$ and $\mathsf{C}_{j_1}, \ldots, \mathsf{C}_{j_s}$, and finally, $\mathcal{Q}$ consists in the binary set $\{0, 1\}$ (*i.e.* the two possible hyperplanes).

The first linear cryptanalysis phase consists in evaluating the bias, or more precisely, *the absolute bias*, as the cryptanalyst ignores the right part of (3), of the linear expression for all possible subkey candidates and for all known plaintext-ciphertext pairs:

$$\Sigma \triangleq \left|\Psi_\ell - \frac{n}{2}\right| \tag{4}$$

[1] In the literature, this non-linearity measure is often called *linear probability*, and expressed as $\mathrm{LP}^f(a,b) \triangleq (2\Pr[a \cdot x = b \cdot f(x)] - 1)^2 = 4\epsilon^2$, where a and b are the masks selecting the plaintext and ciphertext bits, respectively. In this paper, we will refer to the *bias* ϵ for simplicity reasons.

where Ψ_ℓ is the number of times where (3) is equal to 0 (for a given subkey candidate ℓ) and n is the number of known plaintext-ciphertext pairs. In a second phase, the list of subkey candidates is sorted, and the missing key bits are finally searched exhaustively for each subkey candidate until the correct full key is found. The computational complexity of the attack is then related to the number of encryptions needed in the exhaustive search part. The (implicit) mark used to sort the subkey candidates is the following:

Definition 1 (Single-List Ranking Procedure). *The mark μ_ℓ given to a subkey candidate ℓ is defined to be equal to the bias*

$$\mu_\ell \triangleq \Sigma = \left|\Psi_\ell - \frac{n}{2}\right| \tag{5}$$

produced by this subkey ℓ.

Interestingly, the refined version of linear cryptanalysis described in [Mat94] uses *two* biased linear expressions involving *different*[2] key bits subsets. The heuristic proposed by Matsui (which was based on intuition[3]) is the following:

Definition 2 (Double-List Ranking Procedure). *Let $\mathcal{U}_1$ and $\mathcal{U}_2$ be two lists of subkey candidates involving* disjoint *key bits subsets. Sort them independently using the Single-List Ranking Procedure described in Def. 1. Let $\rho^{(\mathcal{U})}(\ell)$ be a function returning the rank of the candidate ℓ in the list $\mathcal{U}$. The* Double-List Ranking Procedure *is then defined as follows:*

1. *To each candidate $\ell = (\ell_1, \ell_2) \in \mathcal{U}_1 \times \mathcal{U}_2$, assign the mark*

$$\mu_{(\ell_1,\ell_2)} \triangleq \rho^{(\mathcal{U}_1)}(\ell_1) \cdot \rho^{(\mathcal{U}_2)}(\ell_2) \tag{6}$$

2. *Sort the "composed" candidates by increasing marks $\mu_{(\ell_1,\ell_2)}$.*

3 An Alternative View on Ranking Procedures

In this section, we recall some well-known statistical hypothesis testing concepts, and we discuss the optimality of the two ranking procedures described above.

---

[2] The different problem consisting in dealing with *multiple linear approximations* has been studied by Kaliski and Robshaw in [KR94]. However, the setting is different than our: they handle the case where one disposes of several linear approximations acting on the *same key bits*, and they compute the *cumulated* (resulting) bias.

[3] Private communication.

3.1 Hypothesis Tests

Let D_0 and D_1 be two different probability distributions defined on the same finite set $\mathcal{X}$. In a *binary hypothesis testing problem*, one is given an element $x \in \mathcal{X}$ which was drawn according either to D_0 or to D_1 and one has to decide which is the case. For this purpose, one defines a so-called *decision rule*, which is a function $\delta : \mathcal{X} \to \{0,1\}$ taking a sample of X as input and defining what should be the guess for each possible $x \in \mathcal{X}$. Associated to this decision rule are two different types of error probabilities: $\alpha \triangleq \Pr_{X \leftarrow \mathsf{D}_0}[\delta(x) = 1]$ and $\beta \triangleq \Pr_{X \leftarrow \mathsf{D}_1}[\delta(x) = 0]$. The decision rule δ defines a partition of $\mathcal{X}$ in two subsets which we denote by $\mathcal{A}$ and $\overline{\mathcal{A}}$, *i.e.* $\mathcal{A} \cup \overline{\mathcal{A}} = \mathcal{X}$; $\mathcal{A}$ is called the *acceptance region of* δ. We recall now the Neyman-Pearson lemma, which derives the shape of the optimum statistical test δ between two simple hypotheses, *i.e.* which gives the optimal decision region $\mathcal{A}$.

Lemma 1 (Neyman-Pearson). *Let X be a random variable drawn according to a probability distribution D and let be the decision problem corresponding to hypotheses $X \leftarrow \mathsf{D}_0$ and $X \leftarrow \mathsf{D}_1$. For $\tau \geq 0$, let $\mathcal{A}$ be defined by*

$$\mathcal{A} \triangleq \left\{ x \in \mathcal{X} : \frac{\Pr_{X \leftarrow \mathsf{D}_0}[x]}{\Pr_{X \leftarrow \mathsf{D}_1}[x]} \geq \tau \right\} \tag{7}$$

Let $\alpha^* \triangleq \Pr_{X \leftarrow \mathsf{D}_0}\left[\overline{\mathcal{A}}\right]$ and $\beta^* \triangleq \Pr_{X \leftarrow \mathsf{D}_1}[\mathcal{A}]$. Let $\mathcal{B}$ be any other decision region with associated probabilities of error α and β. If $\alpha \leq \alpha^*$, then $\beta \geq \beta^*$.

Hence, the Neyman-Pearson lemma indicates that the optimum test (regarding error probabilities) in case of a binary decision problem is the *likelihood-ratio test*. All these considerations are summarized in Def. 3.

Definition 3 (Optimal Binary Hypothesis Test). *To test $X \leftarrow \mathsf{D}_0$ against $X \leftarrow \mathsf{D}_1$, choose a constant $\tau > 0$ depending on α and β and define the likelihood ratio*

$$\mathrm{lr}(x) \triangleq \frac{\Pr_{X \leftarrow \mathsf{D}_0}[x]}{\Pr_{X \leftarrow \mathsf{D}_1}[x]} \tag{8}$$

The optimal decision function is then defined by

$$\delta_{\mathrm{opt}} \triangleq \begin{cases} 0 \ (i.e \text{ accept } X \leftarrow \mathsf{D}_0) \ \text{ if } \mathrm{lr}(x) \geq \tau \\ 1 \ (i.e. \text{ accept } X \leftarrow \mathsf{D}_1) \text{ if } \mathrm{lr}(x) < \tau \end{cases} \tag{9}$$

3.2 The Neyman-Pearson Ranking Procedure

We apply now the Neyman-Pearson paradigm to the ranking procedure. One defines the two hypotheses as follows: H_0 is the hypothesis that the random variable modeling the statistic Σ_ℓ (we make here a slightly abuse of notation by assigning the same name to both entities) produced by a given subkey candidate ℓ is distributed according D_R, *i.e.* it is distributed as the right subkey candidate, while H_1 is the hypothesis that Σ_ℓ follows the distribution D_W, *i.e.* it is distributed as a false subkey candidate (note that we assume here that the "wrong-key randomization hypothesis" [HKM95] holds, *i.e.* that wrong keys follow all the same distribution):

$$H_0 : \Sigma_\ell \leftarrow \mathsf{D}_R$$
$$H_1 : \Sigma_\ell \leftarrow \mathsf{D}_W$$

In this scenario, a type I error (occurring with probability α) means that the correct subkey candidate ℓ_R, with $\Sigma_{\ell_R} \leftarrow \mathsf{D}_R$, is decided to be a wrong one; a type II error (occurring with probability β) means that one accepts a wrong candidate ℓ_W as being the right one.

When performing binary hypothesis tests, one usually proceeds as follows: one chooses a fixed α that one is willing to accept, one computes the threshold τ corresponding to α and one defines the following decision rule when given the statistic Σ_ℓ produced by the candidate ℓ:

$$\begin{cases} H_0 \text{ is accepted if } \frac{f_{\mathsf{D}_R}(\Sigma_\ell)}{f_{\mathsf{D}_W}(\Sigma_\ell)} \geq \tau \\ H_1 \text{ is accepted if } \frac{f_{\mathsf{D}_R}(\Sigma_\ell)}{f_{\mathsf{D}_W}(\Sigma_\ell)} < \tau \end{cases}$$

where f_{D_R} and f_{D_W} denote the density function of the distributions D_R and D_W, respectively.

In our scenario, this means that, given a candidate ℓ, the cryptanalyst will define the threshold τ in such a manner that α is negligible, so that the test will virtually always accept H_0: indeed, accepting a subkey candidate as being the right one and take a wrong decision costs an encryption, while deciding that the right candidate is a wrong one causes the failure of the whole attack. Then, the cryptanalyst can rank the candidates by *decreasing likelihood-ratio values*: the greater the value, the more likely it is to be the looked-for candidate. We call this ranking procedure *Neyman-Pearson Ranking Procedure*:

Definition 4 (Neyman-Pearson Ranking Procedure). *To each candidate ℓ, assign the mark*

$$\mu_\ell \triangleq \frac{f_{\mathsf{D}_R}(\Sigma_\ell)}{f_{\mathsf{D}_W}(\Sigma_\ell)} \tag{10}$$

where Σ_ℓ is the statistic produced by the candidate ℓ, and f_{D_R} and f_{D_W} are the density functions of Σ_ℓ in case of the right and a wrong key, respectively. Then, sort the candidates by decreasing values of μ_ℓ.

Multiple (note that these considerations are valid for more than two lists, too) lists giving information on disjoint subsets of the key bits can thus be optimally combined easily if the *joint distribution* of the underlying statistics is available. Usually, reasonable heuristic statistical independence assumptions can be taken.

We show now that, in case of a linear cryptanalysis, Matsui's single-list ranking procedure is equivalent to a Neyman-Pearson Ranking Procedure. Without loss of generality, we will assume that the linear expression (3) has a bias equal to ϵ (see (1) for a definition of the bias), with $\epsilon > 0$. Approximations of the Σ distributions are known (we refer to [Jun01] for more details about the derivations of these expressions):

$$f_{\mathsf{D}_W}(x) = \sqrt{\frac{8}{n\pi}} e^{-\frac{2x^2}{n}}, \quad \text{for } x \geq 0 \tag{11}$$

and

$$f_{\mathsf{D}_R}(x) = \sqrt{\frac{2}{n\pi}} \left(e^{-\frac{2(x-n\epsilon)^2}{n}} + e^{-\frac{2(x+n\epsilon)^2}{n}} \right) \quad \text{for } x \geq 0 \tag{12}$$

The likelihood-ratio is then given by a straighforward calculation.

Lemma 2. *In the case of a linear cryptanalysis, the likelihood-ratio is given by*

$$\mathrm{lr}(\Sigma_\ell) = e^{-2n\epsilon^2} \cdot \cosh(4\epsilon\Sigma_\ell), \quad \Sigma_\ell \geq 0 \tag{13}$$

We can now state the following result.

Theorem 1. *Matsui's single-list ranking procedure (as defined in Def. 1) is equivalent to a Neyman-Pearson Ranking Procedure and is furthermore optimal in terms of the number of key tests.*

Proof:
This follows easily from the fact that (13) is a monotone increasing function for increasing $\Sigma \geq 0$ and that the type II error probability is monotonly increasing as the likelihood-ratio is decreasing.

$\diamondsuit$

Furthermore, one can easily observe that Matsui's double-list ranking procedure, although very simple, is not a Neyman-Pearson Ranking Procedure, since it is not a total ordering procedure and it does not make use of the whole information given by each subkey candidate (*i.e.* it does not use the experimental bias associated to each candidate, but only their ranks). The first observation leads to some ambiguity in the implementation of Def. 2. For instance, should the combination of two candidates having respective ranks equal to 1 and 4 be searched for the unknown key bits before or after the combination consisting of two candidates having both rank 2 ? In the next section, we illustrate the use of a Neyman-Pearson ranking procedure in the case of a linear cryptanalysis of DES.

4 A Practical Application

Matsui's refined attack against DES [Mat94] makes use of *two* linear expressions involving disjoint subsets of key bits; one is the best linear expression on 14 rounds of DES and is used for deriving the second one using a "reversing trick". Each of them gives information about 13 key bits, the remaining 30 unknown key bits having to be searched exhaustively. We refer to [Mat94] for the detailed description of both linear approximations.

In order to derive a Neyman-Pearson ranking procedure, one has to compute the joint probability distribution of the statistics Σ_{ℓ_1} and Σ_{ℓ_2} furnished by the two linear expressions. As these statistics are dependant of *disjoint* subsets of the key bits, one can reasonably take the following assumption.

Assumption 1. *For each ℓ_1 and ℓ_2, Σ_{ℓ_1} and Σ_{ℓ_2} are statistically independent, where ℓ_1 and ℓ_2 denote subkey candidates involving* disjoint *key subsets.*

A second assumption neglects the effects of *semi-wrong* keys, *i.e.* keys which behave as the right one according to a list only. This is motivated by the fact that, in case of a linear cryptanalysis of DES, the number of such keys is small, and thus their effect on the joint probability distribution is negligible.

Assumption 2. *For each ℓ_1 and ℓ_2, $\mathbf{\Sigma} \triangleq (\Sigma_{\ell_1}, \Sigma_{\ell_2})$ is distributed according either to $\mathsf{D}_R = \mathsf{D}_R^{(1)} \times \mathsf{D}_R^{(2)}$ or to $\mathsf{D}_W = \mathsf{D}_W^{(1)} \times \mathsf{D}_W^{(2)}$, where $\mathsf{D}_R^{(1)}$ and $\mathsf{D}_R^{(2)}$ are the distributions of the right subkey for both key subsets, and $\mathsf{D}_W^{(1)}$ and $\mathsf{D}_W^{(2)}$ are the distributions of a right subkey for both key subsets, respectively.*

Using these two assumptions, the probability density functions defined in (11) and (12), and the fact that the bias of both linear expression is the same and equal to ϵ, one can derive the likelihood-ratio:

$$\mu_{(\ell_1,\ell_2)} = e^{-4n\epsilon^2} \cdot \cosh(4\epsilon\Sigma_{\ell_1}) \cdot \cosh(4\epsilon\Sigma_{\ell_2}) \tag{14}$$

As (14) is not "numerically" convenient to use, we may approximate it using a Taylor development in terms of ϵ, which gives a very intuitive definition of the Neyman-Pearson ranking procedure:

$$\mu_{(\ell_1,\ell_2)} \approx 1 + (8\Sigma_{\ell_1}^2 + 8\Sigma_{\ell_2}^2 - 4n)\epsilon^2 + O(\epsilon^4) \tag{15}$$

Hence, we can note that it is sufficient to rank the subkey candidates by decreasing values of $\Sigma_{\ell_1}^2 + \Sigma_{\ell_2}^2$, *i.e.* the final mark is just the *Euclidean distance* between an unbiased result and a given sample.

We may generalize this result to the case where the biases, which we denote ϵ_1 and ϵ_2, are different in both equations; in this case, the likelihood-ratio is given by

$$\mu_{(\ell_1,\ell_2)} = e^{-2n(\epsilon_2^1+\epsilon_2^2)} \cosh(4\epsilon_1\Sigma_{\ell_1}) \cosh(4\epsilon_2\Sigma_{\ell_2}) \tag{16}$$

A first order approximation is then given by

$$\mu_{(\ell_1,\ell_2)} \approx 1 + 8\Sigma_{\ell_1}^2\epsilon_1^2 + 8\Sigma_{\ell_2}^2\epsilon_2^2 - 2n(\epsilon_1^2 + \epsilon_2^2) \tag{17}$$

which is equivalent to put a grade equal to $\mu_{(\ell_1,\ell_2)} = \Sigma_{\ell_1}^2\epsilon_1^2 + \Sigma_{\ell_2}^2\epsilon_2^2$. We summarize these facts in the following theorem.

Theorem 2. *Under Assumptions 1 and 2, in a linear cryptanalysis using t approximations on disjoint key bits subsets having each a bias equal to $\epsilon_i, 1 \leq i \leq t$, a procedure ranking the subkey candidates by decreasing*

$$\mu_{(\ell_1,\ldots,\ell_t)} = \sum_{i=1}^{t} (\Sigma_{\ell_i}\epsilon_i)^2 \tag{18}$$

is a Neyman-Pearson ranking procedure, and furthermore, it is optimal in terms on key tests.

Sketch of the proof: The proof is similar to the one of Theorem 1 and follows from the fact that β is a monotone increasing function when $\mu_{(\ell_1,\ldots,\ell_t)}$ is decreasing. $\diamondsuit$

4.1 Experimental Results

The Neyman-Pearson ranking procedure described in the previous section has been simulated in the context of 21 linear cryptanalysis of DES, using the data of [Jun01]. The following table summarises our experimental results on the complexity of the exhaustive search part of the attack given 2^{43} known plaintext-ciphertext pairs; we use the following notation: $\mu_{\mathcal{C}}$ denotes the average experimental complexity, $\mathcal{C}_{85\%}$ the maximal complexity given a success probability of 85 %, which is the success probability defined by Matsui in [Mat94], $\mathcal{C}_{\mathrm{med}}$ the median, $\mathcal{C}_{\min}$ and $\mathcal{C}_{\max}$ being the extremal values.

	Matsui's Ranking	Optimal Ranking	Δ
$\log_2 \mu_{\mathcal{C}}$	41.4144	40.8723	-31.32 %
$\log_2 \mathcal{C}_{85\%}$	40.7503	40.6022	-9.75 %
$\log_2 \mathcal{C}_{\mathrm{med}}$	38.1267	36.7748	-60.71 %
$\log_2 \mathcal{C}_{\min}$	32.1699	31.3219	-40.00 %
$\log_2 \mathcal{C}_{\max}$	45.4059	44.6236	-41.86 %

These results lead to following observations:

- The average complexity is decreased by a factor of about 30 %. Actually, the average complexity is not a good statistical indicator for the average behavior of the linear cryptanalysis, because most cases have a far lower complexity and only 3 cases have a complexity greater than the average. Thus, those three cases have a considerable influence on the average complexity and it is worth examining the median behavior.
- A perhaps more significant result is that the median complexity is decreased by a factor of about 60 %. Although one have to be careful with this result because of the small size of the statistical samples number, this value seems to be more accurate regarding the real impact of the improved rule as the average one.
- Although the optimal rule decreases the exhaustive search part complexity on average, "pathological" cases where Matsui's heuristic is better than the Neyman-Pearson ranking procedure can occur. One can explain this by the fact that the Σ densities are sometimes bad approximations of the real ones, several heuristic assumptions being involved.

As the data complexity and the computational complexity of a linear cryptanalysis are closely related, it is possible (and desirable in the context of a known-plaintext attack) to convert a gain in the first category to a gain in the second one: even if we decrease sensibly the number of known plaintext-ciphertext pairs, the complexity will remain within reasonable areas: for instance, given $2^{42.46}$ known plaintext-ciphertext pairs, $\hat{\mathcal{C}}_{85\%} = 2^{44.46}$ DES evaluations, and with only 2^{42} pairs, $\hat{\mathcal{C}}_{85\%} = 2^{46.86}$; these experimental values are summarized in the following table:

Data complexity	$2^{42.00}$	$2^{42.46}$	$2^{43.00}$
Time complexity	$2^{46.86}$	$2^{44.46}$	$2^{40.60}$
Success probability	85 %	85 %	85 %

4.2 Other Attacks

Several published attacks (to the best of our knowledge, all are derived from Matsui's paper) use key ranking procedures or suggest them as potential improvment. In [SK98], Shimoyama and Kaneko use quadratic boolean approximations of DES' S-boxes possessing a larger bias. The first part of their attack consists in a traditional linear cryptanalysis, and thus we can apply our optimal ranking procedure; furthermore, another part of their attack consists also in a sorting procedure using Matsui's heuristic.

In [KM01], Knudsen and Mathiassen show how to modifiy Matsui's attack into a *chosen-plaintexts* attack in order to reduce the needs of pairs. Their attack can also use the "reversing trick", *i.e.* one can apply the same linear characteristic on both encryption and decryption function, in order to derive twice as much key bits. A new time, one could use a key-ranking procedure and our optimal rule to define the order of the subkey candidates during the exhaustive search part.

5 Conclusion

In this paper, we show that considering a statistical cryptanalysis in a hypothesis testing framework allows to define the shape of an *optimal* distinguisher. We note that one can apply such a distinguisher to various published attacks, all of them being more or less related to Matsui's linear cryptanalysis as applied against DES.

We demonstrate experimentally that our distinguisher, in the case of a classical linear cryptanalysis of DES, allows a non-trivial computational complexity decrease. Simulations on 21 real attacks suggest an average complexity of $2^{40.87}$ DES evaluations instead of $2^{41.41}$, as stated in [Jun01].

If one accepts a 15 % failure probability, which is the usual setting, the complexity had upper bound $2^{40.61}$.

Equivalently, as exhaustive search operations are typically less costly than the collection of known plaintext-ciphertext pairs, this technique allows to decrease the number of needed pairs and to keep the computational complexity of the attack in cryptanalyst-friendly areas. Our experiments led, with a success probability of 85 %, to $2^{44.85}$ DES evaluations given $2^{42.46}$ pairs, or to $2^{46.86}$ DES evaluations given only 2^{42} pairs.

Finally, we would like to outline that statistical hypothesis testing concepts seem to be very useful when considering distinguishing procedures in both theoretical and experimental settings. This seems to be confirmed by the increasing interest of the cryptology community in this kind of mathematical tools.

Acknowledgments

We would like to thank Thomas Baignères and the anonymous reviewers for useful and interesting comments.

References

[CHJ02] D. Coppersmith, S. Halevi, and C. Jutla. Cryptanalysis of stream ciphers with linear masking. In *Advances in Cryptology* - CRYPTO *'02*, volume 2442 of *LNCS*, pages 515–532. Springer-Verlag, 2002.

[DES77] National Bureau of Standards. *Data Encryption Standard.* U. S. Department of Commerce, 1977.

[FM01] S. R. Fluhrer and D. A. McGrew. Statistical analysis of the alleged RC4 keystream generator. In FSE *'00*, volume 1978 of *LNCS*, pages 19–30. Springer-Verlag, 2001.

[GM] J.D. Golić and R. Menicocci. Edit probability correlation attacks on stop/go clocked keystream generators. To appear in the Journal of Cryptology.

[HKM95] C. Harpes, G. Kramer, and J.L. Massey. A generalization of linear cryptanalysis and the applicability of Matsui's piling-up lemma. In *Advances in Cryptology* - EUROCRYPT *'95*, volume 921 of *LNCS*, pages 24–38. Springer-Verlag, 1995.

[Jun01] P. Junod. On the complexity of Matsui's attack. In *Selected Areas in Cryptography,* SAC *'01*, volume 2259 of *LNCS*, pages 199–211. Springer-Verlag, 2001.

[Jun03] P. Junod. On the optimality of linear, differential and sequential distinguishers. To appear in *Advances in Cryptology* - EUROCRYPT *'03*, *LNCS*. Springer-Verlag, 2003.

[KM01] L.R. Knudsen and J.E. Mathiassen. A chosen-plaintext linear attack on DES. In FSE *'00*, volume 1978 of *LNCS*, pages 262–272. Springer-Verlag, 2001.

[KR94] B. S. Kaliski and M. J. B. Robshaw. Linear cryptanalysis using multiple approximations. In *Advances in Cryptology* - CRYPTO *'94*, volume 839 of *LNCS*, pages 26–39. Springer-Verlag, 1994.

[Mat93] M. Matsui. Linear cryptanalysis method for DES cipher. In *Advances in Cryptology* - EUROCRYPT *'93*, volume 765 of *LNCS*, pages 386–397. Springer-Verlag, 1993.

[Mat94] M. Matsui. The first experimental cryptanalysis of the Data Encryption Standard. In *Advances in Cryptology* - CRYPTO *'94*, volume 839 of *LNCS*, pages 1–11. Springer-Verlag, 1994.

[Mir02] I. Mironov. (Not so) random shuffles of RC4. In *Advances in Cryptology* - CRYPTO *'02*, volume 2442 of *LNCS*, pages 304–319. Springer-Verlag, 2002.

[MPWW95] S. Murphy, F. Piper, M. Walker, and P. Wild. Likelihood estimation for block cipher keys. Technical report, Information Security Group, University of London, England, 1995.

[SK98] T. Shimoyama and T. Kaneko. Quadratic relation of s-box and its application to the linear attack of full round DES. In *Advances in Cryptology* - CRYPTO *'98*, volume 1462 of *LNCS*, pages 200–211. Springer-Verlag, 1998.

[Vau96] S. Vaudenay. An experiment on DES statistical cryptanalysis. In *3rd ACM Conference on Computer and Communications Security*, pages 139–147. ACM Press, 1996.