Reflection Attacks on Product Ciphers

Orhun Kara

TÜBİTAK UEKAE 41470, Gebze/Kocaeli Turkey
orhun@uekae.tubitak.gov.tr

Abstract. In this paper we describe a novel attack method on product ciphers, which we call the *reflection attack*. The attack method exploits certain similarities among round functions which have not been utilized in previous self similarity attacks. We give practical examples illustrating the power of the reflection attack on several ciphers such as GOST, DEAL and some variants of DES and Magenta. Many interesting and exceptional properties of the method are also presented in these examples. In particular, we demonstrate a known plaintext attack on the full - round GOST, mounted successfully at approximately 73.15 percent of keys. In addition, we discuss new design criteria that make product ciphers resistant to self similarity exploiting attack types and introduce the definition of similarity degree.

Key words: Block Cipher, Round Function, Round Key, Key Schedule, Cryptanalysis, Self Similarity.

1 Introduction

Most prevailing and powerful attacks on product ciphers such as differential attack [3] and linear attack [22] exploit some statistical deviations of round functions that do not disappear after several iterations. Accordingly, a statistical distinguisher for the whole encryption/decryption function is established. In general, the number of rounds increases the workload of such an attack exponentially since the bias diminishes at each iteration. Exceptionally, only two attack methods independent of round numbers have been discovered since the first attempt by Grossman and Tuckerman in 1978 [12]. These are slide attacks [5, 6] and related key attacks [2]. Both of them exploit some degree of self similarity of round functions. In this paper, we introduce a novel attack method, which we call the *reflection attack*[1]. This attack method is also a kind of self similarity analysis. The attack exploits similarities of some round functions of encryption process with those of decryption. Its assumptions are different from those in [5, 6] or in [2] and hence it is applicable to some ciphers resistant to previous self similarity attacks.

We apply the attack on GOST, 2K-DES (a variant of DES defined in [5]), DEAL and MagentaP2 (a new variant of Magenta which is actually expected to be stronger than itself). We introduce an attack on full-round GOST which is the first attack faster than exhaustive search. It works at approximately 73.2 percent of key space with a complexity of 2^{192} steps and 2^{64} known plaintexts. We recover the 2K-DES keys in a shorter time than one encryption step by using 2^{33} known plaintexts. We detect that some certain keys of DEAL are vulnerable to reflection attacks. Their numbers are 2^{16}, 2^{80} and 2^{88}, and they can be recovered in 2^{72}, 2^{136} and 2^{200} steps for 128-bit, 192 bit and 256-bit key lengths respectively. The data complexity is around 2^{66} known plaintext in all cases. Finally, we analyze Magenta and discover a distinguisher. We apply it to attack MagentaP2, a new variant of Magenta which is a double encryption of Magenta including two more rounds. The workloads are $2^{64.8}$, $2^{131.1}$ and 2^{196} encryptions using $2^{65}, 2^{65.6}$ and 2^{66} known plaintexts for

[1] This should not be confused with reflection attack on challenge response authentication system or distribution reflection denial of service. They are subjects of different domains.

128 bit, 192 bit and 256 bit key lengths respectively. Note that MagentaP2 is expected to resist any analyses including the attack in [4] since its number of rounds is $2r + 2$ when Magenta has r rounds.

Reflection attacks have several interesting and unusual properties. We list some of them:

1. Weaknesses of round functions are not exploited in most cases. Hence, the attack works for any round function in these cases. In this paper, we analyze four ciphers. Any weaknesses of round functions are not exploited in any of them.
2. The workload is independent of the number of rounds in some cases. For example, for all the ciphers analyzed here, except DEAL, the workload is independent of round numbers.
3. It is quite unusual that in some cases, increasing the number of rounds may cause weakness in terms of reflection analysis. Magenta is strong against reflection analysis. However, reflection attack works quite well on MagentaP2. Remark that MagentaP2 is expected to resist all other attacks including previous self similarity attacks and the attack of Biham *et. al.* [4]. Another interesting example is GOST. Removing first 8 rounds of GOST results in reducing the number of weak keys by approximately $2^{252.68}$. This amounts to roughly 16 percent of weak keys. These analyses form non-trivial counter examples against the belief that higher number of iterations results in a stronger (at least not weaker) cipher.
4. It is realistic and open to generalizations. The reflection attack is realistic since it breaks actual ciphers and generalization of its assumptions is possible. In this paper, the most trivial similarity exploited is equality. We introduce a novel definition of "similarity degree" which generalizes equality and also diffusion properties of functions. The attack can be applicable to more general class of ciphers in this case. We state some questions in this direction.
5. In some cases, it is more powerful than previous self similarity analyses in terms of both complexity and assumptions. We reduce the complexity of the attack in [5] mounted on 2K-DES. Furthermore, the assumptions are much weaker. If 32 different round keys were used in 2K-DES by repeating in reversing order, then the reflection attack would still work whereas slide attacks [5, 6] would possibly fail. In addition, GOST and MagentaP2 are expected to resist to all known attacks including previous self similarity attacks. These examples exhibit the power of the reflection attack.
6. It is extraordinary for modern ciphers that a component of a cipher designed in order to resist against an attack causes weakness that could be exploited in some other attacks. We give an interesting example: The twist in the order of round keys in last eight rounds of GOST thwarts slide attacks [6]. Existence of the twist is discussed in [6] and it is concluded that GOST is less secure without it. In contrast, it is amazing that the reflection attack exploits twist property of GOST. If the twist were canceled, the reflection attack would probably not work.
7. It is possible to mount reflection attack on ciphers having strong and complicated key schedules. As an illustration, we show existence of weak keys of DEAL. Note that DEAL makes use of the DES encryption for producing each round key.

This paper is organized as follows. We introduce notations and summarize previous self similarity analyses given in [5, 6] and in [2] in Section 2. The fundamental idea of reflection attacks and general statements are given in Section 3, including the assumptions and description of typical attack on Feistel networks. In the following four sections, we give attack examples on four different ciphers. In Section 8, we generalize the attack idea and introduce the definition of self similarity degree of two functions. This new definition can be considered as a generalization of several diffusion criteria such as those given in [13, 35, 29, 33, 23]. Then, several questions related to exploiting more general

forms of self similarity follow. In Section 9, we impose two new security criteria. One of them is about key schedule and the other is on block length. Finally, we conclude the paper by predicting developments in new expansions of self similarity analysis in the last section.

2 Notations and Previous Self Similarity Analyses

Let $E_K : GF(2)^n \rightarrow GF(2)^n$ be an encryption function defined by a key material K and $D_K : GF(2)^n \rightarrow GF(2)^n$ be its inverse mapping. Assume that E_K is a composition of some functions:

$$E_K(x) = F_{k_r} F_{k_{r-1}} \cdots F_{k_1}(x), x \in GF(2)^n,$$

where r is the number of rounds, $k_1, ..., k_r$ are subkeys (round keys) and F_{k_i} is the ith round function. We use these notations throughout the paper.

One of the generic attack methods that exploits some degree of self similarity is the "slide attack"[5, 6]. The typical slide attack can be applied if the sequence of round keys has a too short period, such as 1, 2, 4 etc. For instance, if all the round keys are equal, $k_i = K$, then the encryption function will be $E_K(x) = F_K^r(x) = y$. Let $F_K(x) = x'$. Encrypting x' we have $E_K(x') = y'$. Then, from these two encryptions we obtain two equations which are probably much easier to solve: $F_K(x) = x'$ and $F_K(y) = y'$. Such (x, x') pairs are called *slid pairs*. The laborious part of the attack is to identify slid pairs. This basic attack can be generalized if the period of sequence of round keys is 2 (i.e., $k_i = k_{i+2}$) or 4 (i.e., $k_i = k_{i+4}$) [6].

Related key attacks proposed by Biham [2] are based on a powerful assumption that the attacker knows a relation between several keys and can access encryption function with these related keys. The goal is to find the keys . The most basic type of relation defined over a pair of keys is that the ith subkey of one is equal to the $i + 1$st subkey of the other.

3 Description of The Basic Attack

Define the composite of $j - i + 1$ functions $F_K[i, j]$ starting from i as

$$F_K[i, j] = F_{k_j} \cdots F_{k_i} \text{ for } 1 \leq i < j \leq r \tag{1}$$

and as identity map for $i > j$. Such functions can be called *intermediate functions*. Let $U_K(i, j)$ be the set of fixed points of the function $F_K[i, j]$. More explicitly,

$$U_K(i, j) = \{x \in GF(2)^n : F_K[i, j](x) = x\}.$$

The following statement plays a crucial role in the basic attack:

Theorem 1. *Let i, j be given such that $0 < j - i < i + j \leq r$. Assume that $F_{k_{i-t}} = F_{k_{j+t}}^{-1}$ $\forall t < i$. Then if $F_K[i - t, i - 1](x) \in U_K(i, j)$ then we have $x \in U_K(i - t, j + t)$ for all $t : 1 < t < i$. In addition, if $x \in U_K(i - t, j + t)$ for some $t : 1 < t < i$ then $F_K[i - t, i - 1](x) \in U_K(i, j)$.*

Proof. Assume that $F_K[i - t, i - 1](x) \in U_K(i, j)$. Then we have

$$\begin{aligned} F_K[i - t, j + t](x) &= F_K[j + 1, j + t] \cdot F_K[i, j] \cdot F_K[i - t, i - 1](x) \\ &= F_K[j + 1, j + t] \cdot F_K[i - t, i - 1](x), \text{ since } F_K[i - t, i - 1](x) \in U_K(i, j), \\ &= x, \text{ since } F_{k_{i-t}} = F_{k_{j+t}}^{-1} \text{ for all } t < i. \end{aligned}$$

Hence $x \in U_K(i-t, j+t)$. On the contrary, assume that $x \in U_K(i-t, j+t)$ for some $t : 1 < t < i$. Then the input of $F_K[i,j]$ is $F_K[i-t, i-1](x)$ and the output of $F_K[i,j]$ is $F_K^{-1}[j+1, j+t](x)$. However, $F_K^{-1}[j+1, j+t] = F_K[i-t, i-1]$ since we assume that $F_{k_{i-t}} = F_{k_{j+t}}^{-1}$ for all $t < i$. Hence, $F_K[i-t, i-1](x) \in U_K(i,j)$.

□

One immediate result can be deduced from the theorem above by taking the parameter t in the statement as $i-1$:

Corollary 1. *Assumptions in Theorem 1 imply that $x \in U_K(1, j+i-1)$ if and only if $F_K[1, i-1](x) \in U_K(i,j)$.*

Another corollary can lay the groundwork for an attack on product ciphers whose some round functions in encryption equal some round functions in decryption:

Corollary 2. *Let i, j be given such that $0 < j - i < i + j \leq r$. Assume that $\forall t < i$ we have $F_{k_{i-t}} = F_{k_{j+t}}^{-1}$. Then the encryption function E_K is equal to the function $F_K[i+j, r]$ on the set $F_K^{-1}[1, i-1](U_K(i,j))$.*

Proof. The set $F_K^{-1}[1, i-1](U_K(i,j))$ is equal to $U_K(1, j+i-1)$ by Theorem 1. On the other hand, we have $F_K[1, j+i-1](x) = x$ for $x \in U_K(1, j+i-1)$ by definition. Thus,

$$E_K(x) = F_{k_r} \cdots F_{k_1}(x) = F_K[i+j, r] \cdot F_K[1, i+j-1](x) = F_K[i+j, r](x)$$

□

Corollary 2 states that there exists another function which equals to encryption function on some special subset of the encryption space. This function is probably much weaker then the encryption function since its number of rounds may be much less than r. Then, the attack below can recover the round keys $k_{i+j}, ..., k_r$ by solving the system of equations

$$F_K[i+j, r](x) = E_K(x) = y. \tag{2}$$

This is a typical reflection attack. There are three main parameters which specify the complexity of the attack:

1. m: The number of required pairs (x, y) to solve Equation 2. By solving, we mean a unique solution if m equations are correct and a contradiction (no solution) otherwise.
2. $|U_K(i,j)|$: Cardinality of $U_K(i,j)$.
3. $\Pr(F_K[1, i-1](x) \in U_K(i,j) \,|\, E_K(x))$: The probability that $F_K[1, i-1](x)$ is in $U_K(i,j)$ given $E_K(x)$.

The probability that $F_K[1, i-1](x)$ is in $U_K(i,j)$ is $\frac{|U_K(i,j)|}{2^n}$ for randomly chosen x. However, for given particular values of $E_K(x)$, the probability may be much larger or smaller than $\frac{|U_K(i,j)|}{2^n}$ depending on the structure of a cipher. This structure is crucial in determining time complexity. We have observed that some conditional probabilities are extremely large for some recent ciphers and have successfully mounted reflection attack by exploiting this deviation. For example, the probability $\Pr(F_K[1, i-1](x) \in U_K(i,j) \,|\, E_K(x))$ is one for GOST and one half for 2K-DES. We give the details in the next sections.

Theorem 2. *Assume that we need m pairs to solve Equation 2 and let C be the time, the number of encryptions, required for solving it. Then the attack recovering the round keys $k_{i+j}, ..., k_r$ has a data complexity of $\frac{m \cdot 2^n}{|U_K(i,j)|}$ known plaintexts. Assume the probabilities $\Pr(F_K[1,i](x) \in U_K(i,j) \,|\, E_K(x))$ are pre-calculated and the biggest ℓ probabilities are chosen among $\frac{m \cdot 2^n}{|U_K(i,j)|}$ plaintexts so that*

$$\sum_{s=1}^{\ell} \Pr(F_K[1,i](x_s) \in U_K(i,j) \,|\, E_K(x_s)) \approx m. \tag{3}$$

Then the attack has time complexity bounded above by $\binom{\ell}{m} \cdot C$ encryptions.

Proof. We need m elements of $F_K^{-1}[1,i-1](U_K(i,j))$ for solving Equation 2. The expected number of elements of $F_K^{-1}[1,i-1](U_K(i,j))$ among randomly chosen t plaintexts is $t \cdot \frac{|U_K(i,j)|}{2^n}$. So, t should be approximately $\frac{m \cdot 2^n}{|U_K(i,j)|}$ to get about m elements of $F_K^{-1}[1,i-1](U_K(i,j))$.

Each plaintext ciphertext pair gives an equation like Equation 2. However, only approximately m of them are correct equations. One might try all the plaintexts for solving Equation 2 exhaustively. However, if $\Pr(F_K[1,i](x) \in U_K(i,j) \,|\, E_K(x))$ are large enough for some x, then it is more likely that the corresponding equations are correct. Choose ℓ plaintexts, $x_1, ..., x_\ell$ such that Equation 3 holds. Then, the expected number of correct equations is m among these ℓ equations. The correct equation set can be obtained by trying all subsets of m elements of $\{x_1, ..., x_\ell\}$. But, since the search should be sorted according to the probabilities of subsets we get an upper bound, $\binom{\ell}{m} \cdot C$ for time complexity.

□

Let us note that false alarm probability is disregarded in the theorem since we assume that the solution set is empty if at least one of the equations is incorrect.

Remark 1. In this section, we give only the general idea of the attack. This is a general description and open to straight improvements in some special examples. For instance, the attack is explained only for encryption function. One may repeat the attack for decryption function and improve the complexity. In addition, the success of the attack depends on the number of fixed points of chosen intermediate function. This function does not have to be composite of some consecutive rounds. For instance, it seems an appropriate choice to take the intermediate function as 1.5 rounds in Feistel ciphers (including two swaps instead of one) since 1.5 rounds have many fixed points.

3.1 Reflection Attack on Feistel Network

Let a plaintext $x \in GF(2)^n$ be given as $x = (x_0, x_1); x_0, x_1 \in GF(2)^{n/2}$. The Feistel structure can be stated as a recursive function defined as $x_i = R_{k_{i-1}}(x_{i-1}) \oplus x_{i-2}$ with the initial conditions given by $x = (x_0, x_1)$. The function $R : GF(2)^{n/2} \to GF(2)^{n/2}$ is the encryption function and $\oplus$ is the "XOR" operation. The i-th round operation is defined as

$$(x_i, x_{i+1}) = F_{k_i}(x_{i-1}, x_i) = (x_i, R_{k_i}(x_i) \oplus x_{i-1}) \tag{4}$$

for $i < r$. In general, the swap operation is excluded in the last round and (x_{r+1}, x_r) is the corresponding ciphertext. With some abuse of terminology, R is also called the round function. We call the stream $x_0, x_1, ..., x_r, x_{r+1}$ the *encryption stream* of $x = (x_0, x_1)$ with respect to K.

Proposition 1. *For a given natural number $m < r$, assume that $k_{m-i} = k_{m+i}$, $\forall i : 1 \leq i \leq \min\{r-m, m-1\}$. Let $x = (x_0, x_1)$ be encrypted and $x_0, x_1, ..., x_r, x_{r+1}$ be its encryption stream. If $R_{k_m}(x_m) = 0$ then $x_{m-i} = x_{m+i}$, $\forall i : 1 \leq i \leq \min\{r-m, m-1\}$. Conversely, if $x_{m-i} = x_{m+i}$ and $x_{m-i+1} = x_{m+i-1}$ for some i then $R_{k_m}(x_m) = 0$.*

Proposition 1 had been already known during the studies on cycle structures of DES (see [8, 25]). Hence, the notion of the fix points of the weak keys of DES is well known. However, the studies were focused on algebraic properties of DES permutations and their short cycles rather than developing a key recovery attack [8, 25, 17, 24]. The following corollary points out the opposite direction of this old phenomenon.

Corollary 3. *Assume that each round key k_i determines a round function R_{k_i} randomly. Let $x = (x_0, x_1)$ be encrypted and $x_0, x_1, ..., x_r, x_{r+1}$ be its encryption stream. Assume the round number r is even, $r = 2r'$, and $k_{r'-i} = k_{r'+i}$ $\forall i : 1 \leq i < r'$. Then, $\Pr(x_0 = x_r) = 2^{-\frac{n}{2}+1} - 2^{-n}$ and $\Pr(R_{k_{r'}}(x_{r'}) = 0 \,|\, x_0 = x_r) = \frac{1}{2-2^{-\frac{n}{2}}}$.*

Proof. Assume that the round function is random. Then, the probability that $x_0 = x_r$ is given as

$$\Pr(x_0 = x_r) = 1 \cdot 2^{-n/2} + 2^{-n/2}(1 - 2^{-n/2}) = 2^{-\frac{n}{2}+1} - 2^{-n} \text{ since it is equal to}$$
$$\Pr(x_0 = x_r \,|\, R_{k_{r'}}(x_{r'}) = 0) \Pr(R_{k_{r'}}(x_{r'}) = 0) + \Pr(x_0 = x_r \,|\, R_{k_{r'}}(x_{r'}) \neq 0) \Pr(R_{k_{r'}}(x_{r'}) \neq 0).$$

On the other hand, $\Pr(x_0 = x_r \,|\, R_{k_{r'}}(x_{r'}) = 0) = 1$ by Proposition 1. Hence, we conclude that

$$\Pr(R_{k_{r'}}(x_{r'}) = 0 \,|\, x_0 = x_r) = \frac{\Pr(x_0 = x_r \,|\, R_{k_{r'}}(x_{r'}) = 0) \cdot \Pr(R_{k_{r'}}(x_{r'}) = 0)}{\Pr(x_0 = x_r)} = \frac{2^{-\frac{n}{2}}}{2^{-\frac{n}{2}+1} - 2^{-n}}.$$

□

Corollary 4. *Assumptions are as in Corollary 3. Then the equality $x_0 = x_r$ implies that the following equation is true with probability $\frac{1}{2-2^{-\frac{n}{2}}}$.*

$$x_1 = R_{k_r}(x_r) \oplus x_{r+1}. \tag{5}$$

Proof. Assume that $x_0 = x_r$. Then by Corollary 3 we have $R_{k_{r'}}(x_{r'}) = 0$ with probability $\frac{1}{2-2^{-\frac{n}{2}}}$.Thus the equality $x_1 = x_{r-1}$ is true with probability $\frac{1}{2-2^{-\frac{n}{2}}}$ by Proposition 1. On the other hand $x_{r+1} = R_{k_r}(x_r) \oplus x_{r-1}$. Thus, the probability that $x_1 = R_{k_r}(x_r) \oplus x_{r+1}$ is $\frac{1}{2-2^{-\frac{n}{2}}}$.

□

Reflection Attack on Feistels. Note that the parameters in Corollary 4 are all public except the last round key. (x_0, x_1) forms the plaintext and (x_r, x_{r+1}) forms the corresponding ciphertext. So, Corollary 4 leads to a straightforward attack: Encrypt plaintexts and collect those such that $x_0 = x_r$. If the round keys satisfy that $k_{\frac{r}{2}-i} = k_{\frac{r}{2}+i}$ then the corresponding equations $x_1 = R_{k_r}(x_r) \oplus x_{r+1}$ are correct with probability nearly one half for the collected plaintexts by Corollary 4. Most probably, these equations are easy to solve. Solving them recovers the last round key. One may apply the attack several times with properly chosen parameters or use key schedule for recovering the main key.

4 Cryptanalysis of 2K-DES

2K-DES is one of the modified DES examples given in [5]. 2K-DES uses two independent 48 bit keys K_1 and K_2 and has no key schedule. K_1 is used in the odd rounds and K_2 is used in the even rounds. The total number of rounds is 64. It is most likely that 2K-DES resists to the conventional differential and linear attacks due to its increased number of rounds. Biryukov and Wagner have proposed a slide attack with complexity independent of the number of rounds [5]. The attack uses 2^{32} known plaintexts and its time complexity is 2^{50} 2K-DES encryptions.

Observe that $k_{32-i} = k_{32+i}$ and $k_{33-i} = k_{33+i}$ for $i = 1, ..., 31$. Note that this condition is weaker than that of slide attack in [5]. Hence, one can apply reflection attack to both encryption function and decryption function.

We need to find one plaintext $x = (x_0, x_1)$ satisfying $x_{64} = x_0$ and another plaintext $x' = (x'_0, x'_1)$ satisfying $x'_{65} = x'_1$. The former gives the equation $x_1 = R_{K_2}(x_{64}) \oplus x_{65}$ and the latter gives $x'_{64} = R_{K_1}(x'_1) \oplus x'_0$. Each equation is true with probability nearly one half and one needs approximately 2^{32} known plaintexts to get approximately four equations by Corollary 4. Two equations deduced from $x_{64} = x_0$ will give at most 2^{17} candidates for K_2 whereas other two equations deduced from $x'_{65} = x'_1$ will give at most 2^{17} candidates for K_1. One may get the correct K_1 and the correct K_2 by searching over these solution sets exhaustively. It costs 2^{34} 2K-DES encryptions. As a result the reflection attack on 2K-DES uses 2^{32} known plaintexts and recovers the keys in 2^{34} steps.

It is obvious that the attack can be improved further by increasing the amount of plaintexts. If we use 2^{33} plaintexts then we expect four equations for each key and two of them to be correct. It is most likely that two correct equations out of four give a unique solution and we get no solution for any other two equations. Hence the time complexity is $2 \cdot \binom{4}{2} \cdot C$ by Theorem 2 where $C = 2/64 = 2^{-5}$ encryption. Therefore, time complexity will be less than one.

5 Cryptanalysis of GOST

GOST, the Russian encryption standard [34], is a 32 round 64 bit Feistel network with 256 bit key. It has a simple key schedule: 256 bit key is divided into eight 32 bit words $k_0, ..., k_7$ and the sequence of round keys is given as $k_0, ..., k_7, k_0, ..., k_7, k_0, ..., k_7, k_7, k_6, ..., k_1, k_0$. The round key is included by modular addition in the round function. We do not consider details of the round function. We only assume that it is bijective.

There is no known attack better than exhaustive search. A related key differential cryptanalysis is shown in [19]. The attack is impractical for properly chosen S-boxes with not too bad difference distributions. A slide attack has been mounted on 20 round GOST$\oplus$, a variant of GOST defined in [6]. This attack uses 2^{33} known texts and 2^{65} memory space with 2^{70} encryptions. A recent differential attack given in [31] has been mounted on 21 round GOST and it has data complexity as 2^{56} chosen plaintexts. We propose a basic reflection attack on full round GOST. Its complexity is 2^{64} known plaintext (the whole space) and at most 2^{192} encryptions. Then 256 bit key is recovered.

Take the intermediate function as the 16 round function, $F_{k_0} \cdots F_{k_7} \cdot F_{k_0} \cdots F_{k_7}$. Note that it is also possible to consider the inverse of the intermediate function. This will not cause any change in the notion of the attack. Then we have the following simple fact:

Proposition 2. *A point x is a fixed point of the encryption function if and only if either it is a fixed point of $F_{k_0} \cdots F_{k_7}$ or a point of order two with respect to the function $F_{k_0} \cdots F_{k_7}$ (that is, $F_{k_0} \cdots F_{k_7}(x)^2 = x$).*

Proof. The encryption function and the intermediate function have equal number of fixed points by Theorem 1. On the other hand, observe that a fixed point of intermediate function is either a fixed point of $F_{k_0} \cdots F_{k_7}$ or point of order two with respect to the function $F_{k_0} \cdots F_{k_7}$ (that is, $F_{k_0} \cdots F_{k_7}(x)^2 = x$). However it is straightforward that those points are also fixed points of the encryption function. Hence, such points are the only fixed points of the encryption function. □

This proposition leads to a basic reflection attack: Encrypt all the texts and collect fixed points to a set, say U. For a given $x \in U$ either we have $F_{k_0} \cdots F_{k_7}(x) = x$ or we have $F_{k_0} \cdots F_{k_7}(x) = y$ and $F_{k_0} \cdots F_{k_7}(y) = x$ for some $y \in U$. If U has one element, say x then we have $F_{k_0} \cdots F_{k_7}(x) = x$ which can be used to determine two subkeys by guessing other 6 subkeys. Then check whether a guess is correct by encryption function. If U has two points, say x, y then either we have $F_{k_0} \cdots F_{k_7}(x) = x$ and $F_{k_0} \cdots F_{k_7}(y) = y$ or we have $F_{k_0} \cdots F_{k_7}(x) = y$ and $F_{k_0} \cdots F_{k_7}(y) = x$. In both cases we determine 2 round keys by guessing remaining round keys. The attack is similar when the cardinality of U is bigger than 2.

The success of the attack depends on whether the intermediate function has a fixed point. The probability that a random permutation is a derangement (having no fixed point) is $\frac{1}{e} \approx 36.79\%$ where e is the Euler number. However, the intermediate function is not random since it is a square (square of a permutation) and squares have more fixed points. The probability that a random square is a derangement is about 27%. Hence, slightly more than a quarter of all keys produce derangement intermediate functions in which case the attack can not be mounted. The cardinality of U is expected to be very small, most probably 1, 2 or 3. See the appendix for more explanation and details.

Remark 2. In general, making a product cipher weaker means reducing its number of rounds. Most attacks are mounted on recent ciphers with reduced number of rounds. The attacks given in [19], [6] and [31] on GOST are also mounted on reduced number of rounds. In contrast, this unusual and interesting example of reflection attack on GOST shows that decreasing number of rounds of a cipher does not always mean making it weaker or vice versa. In the example of GOST the intermediate function is a doubling of $F_{k_0} \cdots F_{k_7}$ instead of a random permutation. This increases the number of weak keys by approximately $2^{252.68}$ (10 percent of the key space). The cipher would resist against reflection attacks when used with these about $2^{252.68}$ number of keys if the first eight rounds were removed. Similarly, more copies of $F_{k_0} \cdots F_{k_7}$ forming the intermediate function will result in more fixed points.

Remark 3. Another interesting remark is about the reason of existence of twist in the order of round keys in last eight rounds. The following quotation shows that GOST is believed to be less secure without the twist [6]:

> Why twist? Consider a GOST cipher with a homogeneous key schedule, i.e, omitting the final twist. Is this cipher less secure than GOST? We argue that, if one takes into account the slide attacks, it is...

Indeed, the twist hinders advanced slide attacks. However, it is surprising that reflection attack exploits this twist property.

6 Weak Keys of DEAL

DEAL is a 128 bit block cipher designed by Knudsen [20] and submitted for the AES contest. It is a Feistel network and accepts three different key sizes, namely 128-bit (for 6 rounds), 192-bit (for 6 rounds) and 256-bit (for 8 rounds). DEAL makes use of DES as its round function.

There are some impractical attacks against DEAL. The attack by Knudsen [20] is a meet-in-the-middle attack and requires unrealistically many chosen plaintexts and unrealistic amount of memory. In [21], Lucks uses similar techniques and mounts chosen ciphertext attack on DEAL. A trade-off is given between the number of plaintext/ciphertext pairs and the time complexity. In [18], Kelsey and Schneier discuss the existence of equivalent keys and mount a related key attack. All the attacks require memory and we will not discuss about their workloads.

We mount the reflection attack on DEAL when the key satisfies some conditions . We briefly describe DEAL and explain the attack for 128 bit key-length. The attacks are quite similar for the other cases of key lengths. DEAL-128 uses 128 bit key K, divided into two 64-bit parts as K_1 and K_2. The six round keys, $RK_1, ..., RK_6$, are computed by using DES as $RK_i = E_C(K_{(i \bmod 2)+1} \oplus RK_{i-1} \oplus s_i)$ where E is DES encryption, C is a 56-bit public constant used as a DES key in the key schedule and $RK_0 = 0$. Here s_i's are 64-bit constants. Only 56 bits of each RK_i is used in the i-th round of DEAL which we denote $RED(RK_i)$ (reduction of RK_i to 56 bits). Note that the final round ends with a swap.

Assume that $RED(RK_2) = RED(RK_6)$ and $RED(RK_3) = RED(RK_5)$. The probability that these equalities hold is roughly 2^{-112}. In this case, the last five rounds of DEAL has 2^{64} fixed points (without the last swap). Applying the reflection attack similar to 2K-DES, we obtain around eight equations for the first round encryption by collecting the plaintexts whose left parts are equal to the left parts of their corresponding cipher texts among 2^{66} known plaintexts. This will be enough to decide that the equalities $RED(RK_2) = RED(RK_6)$ and $RED(RK_3) = RED(RK_5)$ hold since otherwise, we would expect around four plaintexts whose left parts are equal to the left parts of their corresponding cipher texts.

Four of the eight equalities are expected to come from fixed points. Hence, we can recover 56 bits of RK_1 by making search over all possible values of $RED(RK_1)$ and checking whether around four of the equations, $E_{RK_1}(x) = y$ hold. Recovering 56 bits of RK_1 yields 56- bit information about the first 64 bit part of the main key, K_1. The remaining unknown key bits may be obtained by applying several attacks on 5-round DEAL (see [20, 21]). However, the simplest way is just making search on remaining bits. So, the time complexity is around 2^{72} steps.

We have the same data complexity for DEAL-192 and DEAL-256. On the other hand, the time complexities are around 2^{136} and 2^{200} steps respectively (this is the complexity of searching remaining bits after recovering 56 bits of a key). Note that we have three equalities instead of two when the key length is 256 bits. Hence, the probability that the equalities hold is roughly 2^{-168} in this case.

7 Cryptanalysis of MagentaP2

Magenta is a block cipher submitted for the AES contest by Deutsche Telekom AG [16]. It is a Feistel cipher with 128 bit block size and 128, 192 or 256 bit key sizes. In this section we give a high level description of Magenta and construct a distinguisher for the whole cipher. This distinguisher does not assist key recovering. We modify Magenta and call it MagentaP2 (meaning Magenta Plus 2). MagentaP2 is double encryption of Magenta plus two more rounds. The modified Magenta is

expected to be more secure then Magenta against most of the attack methods including the attack in [4] on Magenta. However, it is surprising that MagentaP2 is weaker than Magenta itself in terms of reflection attacks.

We give a short description of Magenta. We do not enter into details of round function since we do not exploit it in cryptanalysis. When the key length of Magenta is of 128, 192 or 256 bits then it is divided into two, three or four equal parts as (K_1, K_2), (K_1, K_2, K_3) or (K_1, K_2, K_3, K_4) respectively. The encryption functions are

$$E_K = \begin{cases} F_{K_1}F_{K_1}F_{K_2}F_{K_2}F_{K_1}F_{K_1} & \text{if key size is 128,} \\ F_{K_1}F_{K_2}F_{K_3}F_{K_3}F_{K_2}F_{K_1} & \text{if key size is 192,} \\ F_{K_1}F_{K_2}F_{K_3}F_{K_4}F_{K_4}F_{K_3}F_{K_2}F_{K_1} & \text{if key size is 256.} \end{cases}$$

Each round function F_{K_i} is defined as

$$\begin{aligned} F_{K_i} &: GF(2)^{128} \longrightarrow GF(2)^{128} \\ F_{K_i}(x, y) &= (y, R_{K_i}(y) \oplus x). \end{aligned} \tag{6}$$

Magenta was cryptanalyzed during the AES conferences by Biham *et. al.* [4] and hence eliminated. The attack is a divide and conquer type attack. One can extract the outer keys, independently from the inner key. The complexity is 2^{l_k-31} encryptions for a known plaintext attack where l_k is the key length.

7.1 Description of MagentaP2 and Reflection Attack

Define an intermediate function

$$\begin{aligned} I_{K_i} &: GF(2)^{128} \longrightarrow GF(2)^{128} \\ I_{K_i}(x, y) &= (R_{K_i}(R_{K_i}(y) \oplus x) \oplus y, R_{K_i}(y) \oplus x). \end{aligned} \tag{7}$$

The function I_{K_i} is indeed two rounds of encryption with key K_i such that the second swap is ignored: I_{K_i} is $F_{K_i}F_{K_i}$ without the last swap. We use this function as the intermediate function. It has many fixed points:

Lemma 1. *The function I_{K_i} has 2^{64} fixed points.*

Proof. The fixed points of the function I_{K_i} are those $(x, y) \in GF(2)^{128}$ such that

$$x = R_{K_i}(R_{K_i}(y) \oplus x) \oplus y \text{ and } y = R_{K_i}(y) \oplus x. \tag{8}$$

These are the same equations and the points $(R_{K_i}(y) \oplus y, y)$ are fixed points of I_{K_i} $\forall y \in GF(2)^{64}$. □

We obtain a distinguisher for Magenta according to the proposition. That is, it has 2^{64} fixed points. This distinguisher does not depend on the round number. However, it is difficult to use it to develop a key recovery attack. We have modified Magenta by adding extra rounds. In general, the increased number of rounds is expected to strengthen the cipher, but the situation is unusually converse in terms of reflection analysis: The basic reflection attack works on the modified Magenta.

The modified Magenta, called MagentaP2 is a double encryption of Magenta including two more rounds. Let $E_K^{(M)}$ and $E_K^{(MP2)}$ denote the encryption functions of Magenta and MagentaP2 respectively. Then MagentaP2 encryption is defined as

$$E_K^{(MP2)}(x) = F_{(K_t \ll_m)} E_K^{(M)} E_K^{(M)} F_{K_t}(x) \tag{9}$$

where F is the round function of Magenta and

$$K_t = \begin{cases} K_2 & \text{if key size is 128,} \\ K_2 \oplus K_3 & \text{if key size is 192,} \\ K_2 \oplus K_3 \oplus K_4 & \text{if key size is 256.} \end{cases}$$

$\ll_m$ is cyclic rotation to left by m bits where m can be chosen any positive integer less then 64. The new cipher depends on m but we call all the ciphers simply as "MagentaP2" by abuse of terminology.

The intermediate function of MagentaP2 chosen as

$$I_{K_1}(x, y) = (R_{K_1}(R_{K_1}(y) \oplus x) \oplus y, R_{K_1}(y) \oplus x) \tag{10}$$

also has 2^{64} fixed points by Lemma 1. If the first half of a plaintext is equal to first half of its corresponding ciphertext through encryption of Magenta, then the other halves are also equal with probability nearly one half by Corollary 4. This distinguisher does not depend on the number of Magenta encryptions. Composite of several Magenta functions $E_K^{(M)}$ has the same property and it is used to attack MagentaP2.

The reflection attack on MagentaP2 is to get an equation similar to Equation 5 and solve it to extract the subkey K_t. The following proposition leads to a reflection attack on MagentaP2.

Proposition 3. *Assume that Magenta is a random function. Let a plaintext $x = (x_0, x_1)$ be encrypted by MagentaP2 and the ciphertext $y = (y_0, y_1)$ be obtained. Assume that $x_1 = y_1$. Then x and y satisfy the equation*

$$R_{K_t}(x_1) \oplus R_{K_t \ll_m}(y_1) = x_0 \oplus y_0. \tag{11}$$

with probability $\frac{1}{2-2^{-64}}$.

Proof. Observe that the equations $R_{K_t}(x_1) \oplus R_{K_t \ll_m}(y_1) = x_0 \oplus y_0$ and $x_1 = y_1$ together come from a fixed point $(R_{K_t}(x_1) \oplus x_0, x_1)$ of double encryption Magenta function $E_K^{(M)} E_K^{(M)}$. We have the equality of probabilities:

$$\Pr(F_{K_t}(x) \text{ is fixed point} \,|\, x_1 = y_1) = \frac{\Pr(F_{K_t}(x) \text{ is fixed point})}{\Pr(x_1 = y_1)}$$

since $\Pr(x_1 = y_1 \,|\, F_{K_t}(x) \text{ is fixed point}) = 1$. On the other hand, $\Pr(x_1 = y_1) = 2^{-63} - 2^{-128}$ by Corollary 4 and the result follows.

□

Equation 11 leads to a divide and conquer type attack that can be mounted on MagentaP2. Encrypt a plaintext $x = (x_0, x_1)$ and obtain the corresponding ciphertext $y = (y_0, y_1)$. If $x_1 = y_1$ then Equation 11 is satisfied for x and y with probability nearly one half. Solve the equation and

extract the subkey K_t and then recover the remaining key bits by searching exhaustively. Let the key length be $64 \cdot i$ for $i = 2, 3, 4$. Then by using $i \cdot 2^{64}$ plaintexts we obtain approximately $2i$ equations of the form Equation 11 and expect half of them to be correct by Proposition 3. By collecting the subsets of i equations and solving them we obtain a unique solution for K_t. Note that false alarm probability is almost zero since the probability that a false key is a solution of i equations is 2^{-64i} whereas the space of K_t has $2^{64(i-1)}$ elements. The time complexity of recovering K_t is $\binom{2i}{i}\frac{i \cdot 2^{64i-63}}{r}$ by Theorem 2 where r is the number of rounds, namely 14 or 18 depending on the key size. The remaining key material (i.e, K_1) can be deduced by exhaustive search. As a result, one can recover the key by $2^{64.78}, 2^{131.1}$ and $2^{196.96}$ encryptions using $2^{65}, 2^{65.58}$ and 2^{66} known plaintexts for 128 bit, 192 bit and 256 bit key lengths respectively.

Remark 4. The algorithm Magenta is doubled in the modified version. Indeed, the number of Magenta encryption does not affect the attack complexity. Therefore, one may use triple Magenta encryption or more. Still, the attack will work. It is also interesting that other self similarity attack methods whose complexities are independent of round number, such as related key attacks or slide attacks probably do not work for MagentaP2.

8 Generalization and Questions

We give a novel definition which can be considered as a benchmark for similarity degree.

Definition 1. *Let $F_1, F_2 : GF(2)^n \to GF(2)^m$ be two functions. Then F_1 and F_2 are called similar of degree (d_1, d_2) with probability p if the number of ordered pairs $(x, x') \in GF(2)^n \times GF(2)^n$ satisfying*

$$HW(x \oplus x') \leq n - d_1 \Rightarrow HW(F_1(x) \oplus F_2(x')) \leq m - d_2$$

is $p \cdot 2^n \cdot \sum_{i=0}^{n-d_1} \binom{n}{i}$ where $HW()$ is the Hamming Weight of binary vectors [2].

This definition generalizes the equality of functions. Two functions are equal if and only if they are similar of full degree with probability one. Note that this is also a generalization of several criteria on diffusion of a single function such as those in [13, 35, 29, 33, 23]. A function F is self similar (similar to itself) of degree (d_1, d_2) with probability p means changing $n - d_1$ or less number of bits of an input would cause a change of at most $n - d_2$ bits of its corresponding output with probability p.

Likewise, we can generalize the notion of fixed points of a function in the following definition:

Definition 2. *Let $F : GF(2)^n \to GF(2)^n$ be a function. The points $x \in GF(2)^n$ satisfying $HW(x \oplus F(x)) \leq d$ are called semi-fixed points of degree d.*

Remark that semi-fixed points of degree d are also semi-fixed points of degree d' for $d \leq d'$.

The assumptions of Theorem 1 can be extended by using the definitions of similarity and semi-fixed notions. Thus, we can obtain a statement with generalized assumptions. However, the corresponding reflection attack may be much weaker since similarity probability is expected to diminish at each iteration. On the other hand, similarity may be high with high probability in some subsets of a key space which leads to a weak key space with respect to reflection attacks.

[2] This notion may be considered as a generalization of Lipschitz condition.

The most interesting generalization of the reflection attack may be combining the attack with several statistical attack methods such as differential attacks and linear attacks. Another interesting question is whether reflection attacks can be mounted on SPN structures or stream ciphers.

It is expected that more ciphers having self similarity of certain degree will be susceptible to reflection attacks. Nevertheless, this is a question whether there exists ciphers on which generalized reflection attacks work but basic reflection attacks do not work.

9 New Security Criteria

Some special similarities of round functions can be considered as weakness. Hence, round functions should be independent of each other. In particular, involutions may cause weaknesses. However, they have been accepted as a design criteria for both security and efficiency so far. Thus, it may be inconvenient to refuse such kind of designs. On the other hand, self similarity should be avoided in such designs. As a conclusion, the best way to destroy self similarities of round functions seems imposing security criteria on key schedules.

Some security criteria have been imposed on lengths of parameters of a stream cipher such as IV length [15] and internal state size [1, 11]. The corresponding criterion on block ciphers is that block length should be at least as large as key length if it is operated in a stream mode in order to supply resistance to tradeoff attacks. This is necessary also against distinguishing attacks. Besides, observe that relatively much smaller block length of GOST is also exploited in the reflection attack. These phenomena impose such a security criterion on block length.

Round keys should look random and independent of each other. If round keys are randomly chosen and look "independent" (no dependency can be discovered by a polynomial time test) then the round functions are expected to be "independent" of each other. Some classifications of key schedules have been proposed in [7, 14] according to independence degree of round keys. It was argued that AES key schedule was surprisingly poor and a new key schedule was proposed for AES in [14]. The proposal supplies more "independence" among round key bits, making concession in key agility.

The functions producing round keys can be tested whether they are similar of degree $(d_1, d_2$ with large d_1 and d_2 as a "pseudo-independence" test. A poor key schedule has round key producing functions which are highly similar (of high degree) with high probability. For instance, the key scheduling process of Blowfish is complex (see [30]) but, the slide attack works in some special cases [5]. This is due to high degree of similarity of round key producing functions even though these functions themselves are highly complicated and nonlinear.

The criteria on key schedule compromise key agility. Typical examples are "secure" but heavy key schedules of Twofish, RC6 and Mars among AES finalists. Round keys are produced in 23.3, 6.3 and 5.8 encryptions respectively [9]. Moreover, the criterion on block length would cause decreasing efficiency of block ciphers, compared to stream ciphers. This may support the argument for identifying new stream ciphers through some international research projects such as e-STREAM [10] and NESSIE [26].

10 Conclusion

We have studied self similarity of iterations of product ciphers and given a novel and interesting attack example exploiting self similarity which we called the reflection attack. We have mounted the

attack on 2K-DES, GOST and MagentaP2. In particular, we have decreased the effective security level which the GOST cipher supplies by approximately 64 bits. We have stated some questions about generalization of reflection attacks in terms of its assumptions. In addition, we have defined similarity degree and it turns out that ciphers whose component functions are similar up to certain degree are suspicious to be vulnerable to the attacks. Consequently, we have developed new security criteria about iteration functions.

Most conventional attacks exploit statistical biases of round functions and hence could be rendered useless by supplying enough number of iterations. Differential cryptanalysis [3] and linear cryptanalysis [22] are very well known examples of these statistical attacks. Both methods have imposed several security criteria on round functions without caring about self similarity. For example, the confusion layer of a round function is expected to be highly nonlinear function [22] and have very low δ (differential) uniformity [27]. These security criteria have been taken into consideration in the design of most of the ciphers after these attacks. AES is the most famous example.

These criteria are not related to key schedules since some defined similarities between round functions do not affect the workloads of such attacks. On the other hand, it is most probably that self similarity analyses are open to improvements and generalizations in several directions. Therefore, designers should consider not only security criteria imposed on round functions themselves but also self similarity of these functions.

Acknowledgments

I thank Hüseyin Demirci, Nezih Geçkinli, İsmail Güloğlu, Atilla Arif Hasekioğlu, Cevat Manap and Ali Aydın Selçuk for their constructive comments.

References

1. S. Babbage, *Improved Exhaustive Search Attacks on Stream Ciphers*, European Convention on Security and Detection, IEE Conference publication No. 408, pp. 161-166, IEE, 1995
2. E. Biham, *New Types of Cryptanalytic Attacks Using Related Keys*, J. of Cryptology, Vol.7, pp.229-246, 1994.
3. E. Biham, A. Shamir, *Differential Cryptanalysis of Data Encryption Standard*, Springer Verlag, 1993.
4. E.Biham, A. Biryukov, N. Ferguson, L.R. Knudsen, B. Schneier, A. Shamir, *Cryptanalysis of Magenta*, Second AES conference, 1999.
5. A. Biryukov, D. Wagner, *Slide Attacks*, Proceedings of FSE'99, LNCS 1636, pp.245-259, Springer Verlag, 1999.
6. A. Biryukov, D. Wagner, *Advanced Slide Attacks*, Advances in Cryptology- EUROCRYPT 2000, LNCS 1807, pp.589-606, Springer Verlag, 2000.
7. G. Carter, E. Dawson, L. Nielsen, *Key Schedules of Iterated Block Ciphers*, Proceedings of Information Security and Privacy, ACISP'98, LNCS 1438 pp. 80-89, Springer Verlag, 1998.
8. D. Coppersmith, *The Real Reason for Rivest's Phenomenon*, Advances in Cryptology- CRYPTO'85, LNCS 218, pp. 535-536, Springer Verlag, 1985.
9. B. Gladman, *AES Second Round Implementation Experience*, In Proceedings from Second AES Candidate Conference, NIST, 1999.
10. eSTREAM, Stream Cipher Project, eCRYPT, `http://www.ecrypt.eu.org/stream`.
11. J. Golić, *Cryptanalysis of Alleged A5 Stream Cipher,*Advances in Cryptology- EUROCRYPT'97, LNCS 1233, pp.239-255, Springer Verlag, 1997.
12. E.K. Grossman, B. Tuckerman, *Analysis of a Weakened Feistel-like Cipher*, International Conference on Communications, pp. 46.3.1-46.3.5,Alger Press Limited, 1978.
13. R. Forre, *The Strict Avalanche Criterion: Spectral properties of Boolean Functions and Extended Definitions*, Advances in Cryptology- CRYPTO'88, LNCS 403, pp.450-468, Springer Verlag, 1988.
14. M. Henricksen, *Design, Implementation and Cryptanalysis of Modern Symmetric Ciphers*, PhD Thesis, ISRC, Faculty of Information Technology, Queensland University of Technology, 2005.

15. J. Hong, P. Sarkar, *Rediscovery of the Time Memory Tradeoff*, Cryptology ePrint Archieve, Report 2005/090, 2005.
16. M.J. Jacobson, Jr., K. Huber, *The Magenta Block Cipher Algorithm*, First AES conference, 1998.
17. B.S. Kaliski, R.L. Rivest, T. Sherman, *Is DES a Pure Cipher? (Results of More Cycling Experiments on DES)*, Advances in Cryptology- CRYPTO'85, LNCS 218, pp. 212-222, Springer Verlag, 1985.
18. J. Kelsey, B. Schneier, *Key-Schedule Cryptanalysis of DEAL*, Proceedings of SAC'99, pp.118-134, Springer Verlag, 2000.
19. J. Kelsey, B. Schneier, D. Wagner, *Key-Schedule Cryptanalysis of IDEA, G-DES, GOST, SAFER, and Triple-DES*, Advances in Cryptology, CRYPTO'96, PP 237-251, 1996.
20. L. Knudsen, *DEAL - a 128-Bit Block Cipher*, `http://www.ii.uib.no/ larsr/aes.html`.
21. S. Lucks, *On the Security of 128-Bit Block Cipher DEAL*, Proceedings of FSE'99, LNCS 1636, pp.60-70, Springer Verlag, 1999.
22. M. Matsui, *Linear Cryptanalysis Method of DES Cipher*, Advances in Cryptology- EUROCRYPT 1993, LNCS 765, pp. 386-397, Springer Verlag, 1994.
23. W. Meier, O. Staffelbach, *Nonlinearity Criteria for Cryptographic Functions*, Advances in Cryptology- EUROCRYPT 1989, LNCS 434, pp. 548-562, Springer Verlag, 1989.
24. J.H. Moore, G.J. Simmons, *Cycle Structure of the DES with Weak and Semi-Weak Keys*, Advances in Cryptology-CRYPTO'86, LNCS 263, pp.9-32, Springer Verlag, 1986.
25. J.H. Moore, G.J. Simmons, *Cycle Structure of the DES for Keys Having Palindromic (or Antipalindromic) Sequences of Round Keys*, IEEE Transactions on Software Engineering, pp. 262-273, 13 (1987).
26. NESSIE, New European Schemes for Signatures, Integrity, and Encryption, `http://www.cryptonessie.org.`
27. K. Nyberg, *Differentially Uniform Mappings for Cryptography,* Advances in Cryptology- EUROCRYPT'93, LNCS 765, pp.55-63, Springer Verlag, 1994.
28. J. Riordan, *An Introduction to Combinatorial Analysis*, New York, Wiley, 1958.
29. B. Preneel, W.V. Leekwijck, L.V. Linden, R. Govaerts, J. Vandewalle, *Propagation Characteristics of Boolean Functions*, Advances in Cryptology- EUROCRYPT 1990, LNCS 473, pp. 161-173, Springer Verlag, 1990.
30. B. Schneier, *Description of a New Variable - Length Key, 64 Bit Block Cipher (Blowfish),* Proceedings of FSE'94, LNCS 809, pp.191-204, Springer Verlag, 1994.
31. H. Seki, T. Kaneko, *Differential Cryptanalysis of Reduced Rounds of GOST*, Proceedings of SAC 2000, LNCS 9743, pp. 315-323, Springer Verlag, 2000.
32. S. Vaudenay, *On the Weak Keys of Blowfish*, Proceedings of FSE'96, LNCS 1039, pp.27-32, Springer Verlag, 1996.
33. A.F. Webster, S.E. Tavares, *On the Design of S-boxes*, Advances in Cryptology- CRYPTO'85, LNCS 218, pp.523-534, Springer Verlag, 1985.
34. I.A. Zabotin, G.P. Glazkov, V.B. Isaeva, *Cryptographic Protection for Information Processing Systems. Cryptographic Transformation Algorithm*, Government Standard of the USSR, GOST 28147-89, 1989.
35. Y. Zheng, X.M. Zhang, *GAC- The Criterion for Global Avalanche Characteristics of Cryptographic Functions*, Journal for Universal Computer Science, 1(5), pp. 316-333, 1995.

A Fixed Points of Random Permutations and Squares

Let $\pi \in S_n$ be a random permutation (a permutation chosen randomly from the set of permutations) of the symmetric group S_n. Then the probability that it is a derangement (a permutation having no fixed point) is given by the formula

$$\sum_{i=0}^{n} \frac{(-1)^i}{i!} \approx \frac{1}{e}. \tag{12}$$

This formula comes from an example of the inclusion exclusion principle which gives the number of derangements:

$$D(n) = \sum_{i=2}^{n} \binom{n}{i} (-1)^i (n-i)!. \tag{13}$$

One immediate consequence is that the probability that a random function has a fixed point is approximately $\frac{e-1}{e} \approx 0.6321$. One can count the number of permutations having at least two fixed points by a similar argument and gets that

$$\sum_{i=2}^{n} \binom{n}{i} (-1)^i (i-1)(n-i)! \tag{14}$$

and the probability that it has at least two fixed points will be

$$\sum_{i=2}^{n} \frac{(-1)^i (i-1)}{i!} \approx 26.42\%. \tag{15}$$

Similarly, we have the probability that a random permutation has more than two fixed points is 8.03%. So, if a permutation has fixed points, then the number of fixed points is most probably 1 or 2 (with probability 87.3%). Indeed, a random permutation of length at least m contains on average $\frac{1}{m}$ cycles of length m. So, the average number of fixed points is one and if we exclude derangements then the average will be $1/0.6321 \approx 1.58$. See [28] for detailed information on fixed points.

A permutation which is a square of some permutations is called a square permutation. The probability that a random square permutation has a fixed point is slightly more than 63.21%. Taking squares permutes elements of each odd cycle whereas even cycles split up into two cycles of half the length. So, fixed points remain fixed and any 2- cycle (cycle of length two) splits up into two fixed points. This is why squares have more fixed points.

The question is the amount of increase in the probability of having fixed points in the case of random square permutations. It may be calculated by counting the number of derangements having 2-cycles in their cycle decompositions.

Theorem 3. *The probability that a random square in S_n has a fixed point is approximately* 9.94% *bigger than that of a random permutation in S_n for sufficiently large n.*

Proof. The number of derangements having 2-cycles in their cycle decompositions is given as

$$\sum_{i=1}^{\lfloor n/2 \rfloor} \binom{n}{2i} \binom{2i}{2} (-1)^{i+1} D(n-2i) = \sum_{i=1}^{\lfloor n/2 \rfloor} \sum_{k=2}^{n-2i} \binom{n}{2i} \binom{n-2i}{k} \binom{2i}{2} (-1)^{i+k+1} (n-2i-k)! \tag{16}$$

which leads to an increment in the probability of having some fixed points as

$$\frac{1}{2} \sum_{i=1}^{\lfloor n/2 \rfloor} \sum_{k=2}^{n-2i} \frac{(-1)^{i+k+1}}{k! \cdot (2i-2)!}. \tag{17}$$

On the other hand, this sum tends to 9.94% very fast as n gets bigger. □

So, we have concluded that the probability that a random square has a fixed point is approximately 73.15% whereas the corresponding probability of a random permutation is 63.21%.