Diffusion Behaviour of Cryptographic Primitives in Feistel Networks

Vasilios Katos

Department of Information Systems and Computer Applications,
University of Portsmouth,
Burnaby Terrace, Portsmouth, PO1 3AE

Abstract. The concept of product encryption is resident in the majority of symmetric block ciphers. Along with product encryption, two properties were also defined by Shannon, namely diffusion and confusion. In a product cipher such as a Feistel Network (FN), or generally a Substitution Permutation Network (SPN), diffusion is dependent upon two types of primitives, the nonlinear transformation and the swapping scheme. Different approaches to diffusion analysis considered either the topology of a FN, or the nonlinear transformation. This paper describes a metric for diffusion in a way suitable for investigating the behaviour of the underlying primitives of a FN.

1 Introduction

Since their invention, Feistel Networks (FNs) [1], [2] have been extensively studied and analysed [3], [4]. The large research interest in FNs was due to several reasons:

- flexibility of the underlying non-linear primitive. The main non-linear function involved in a FN, which is not required to be injective, in order to allow unambiguous decryption;
- realisation of product encryption. FNs are excellent examples of product encryption. The concept of product encryption, introduced in [5], states that a chain encryption of “weak” ciphers results into a much stronger one. In the same paper, the notion of confusion and diffusion was introduced, which relate to the cryptographic qualities of a cipher;
- the DES [6], which is probably the most analysed cipher, is a FN.

However, the bulk of the research in FNs is on homogeneous balanced FNs [3], since the DES falls into this category. As a direct consequence, the research interest focused on the construction and properties of the underlying non-linear function. In [3] there is an investigation of the topology of a FN rather than the non-linear function. In the same paper, confusion and diffusion were put into perspective and metrics such as the diffusion rate and confusion rate where defined. A similar perspective is in [4], but the methodology for examining the diffusion involved directed graphs. However, although that a graph is an effective tool, the diffusion capability of a cipher is not apparent as the complexity increases.

Katos V. (2004).
Diffusion Behaviour of Cryptographic Primitives in Feistel Networks.
In *Proceedings of the 2nd International Workshop on Security in Information Systems*, pages 79-87
DOI: 10.5220/0002661300790087

The contribution of this paper is two-fold. First, it provides a step towards an algebraic description of the diffusion capacity of a FN round. This would allow investigation of a much broader category of FNs, namely the unbalanced heterogeneous FNs. Second, the proposed approach allows assumptions about the non-linear function which can be experimentally evaluated. To demonstrate this, a randomness test is described and can be used for evaluating the behaviour of the FN as a pseudorandom function [7],[8].

2 Diffusion instances and diffusion matrix

The idea behind the construction of the diffusion instances is related to the calculation of the differential characteristic, which is the centrepiece of differential cryptanalysis [9]. A block cipher can be viewed as a function with two independent input variables, namely the plaintext (or ciphertext) and the encrypting (or decrypting) key, and one dependent output variable, the ciphertext (or plaintext).

Diffusion is the property where a given input plaintext bit has the chance to affect the output bits [5]. The higher the diffusion, the more output bits can be affected by a certain input bit. In the described method, the diffusion instance is defined. The diffusion instance is a *snapshot* of the diffusion capacity of a cipher.

The process for generating the diffusion instance is similar to the bitwise calculations used for the Strict Avalanche Criterion (SAC) investigation [10]. Given a random plaintext $p_0 \in_U GF(2)^n$ and a nonzero vector $\alpha = (1\ 0\ 0\ ...\ 0)$, we compute:

$$\psi_j = e_k(p_0) \oplus e_k(p_0 \oplus (\alpha \gg j)),\ 0 \leq j \leq n-1 \tag{1}$$

where $(\alpha \gg j)$ represents the right shift of α by j bits.

If $a[k]$ denotes the k-th bit of the binary string a, then matrix Ψ is defined as:

$$\mathbf{\Psi} = \begin{bmatrix} \psi_1[0] & \psi_1[1] & \dots & \psi_1[n-1] \\ \psi_2[0] & \psi_2[1] & \dots & \psi_2[n-1] \\ \vdots & \vdots & \vdots & \vdots \\ \psi_n[0] & \psi_n[1] & \dots & \psi_n[n-1] \end{bmatrix} . \tag{2}$$

The matrix Ψ would then be one diffusion instance. According to the definitions of the characteristics of confusion and diffusion, for a cipher these characteristics are at maximum if a (binary) swap of any of the input bits results to a swap of the output bits with probability of 0.5 for every output bit. The diffusion instance represents the ability of an input bit to affect an output bit, [11].

The diffusion matrix is calculated from the logical OR of the Ψ matrices:

Definition 1. *Let Ψ_i, $i = 1, 2, ...$ be the diffusion instances of a FN. The diffusion matrix is defined as:*

$$\mathcal{D} = \bigvee_i \Psi_i\ . \tag{3}$$

Theoretically, in order to obtain the actual diffusion matrix of a FN, all plaintexts must be considered. In practice, for a FN with a 64 bit input, it appeared that 10 random plaintexts (and therefore 10 diffusion instances, accounting to a total of 640 plaintexts) would suffice for determining the diffusion matrix. More analytically, after combining 10 diffusion instances, there was no change in the resulting diffusion matrix with each additional diffusion instance. Furthermore, for a block cipher with maximum diffusion capabilities, all entries of its diffusion matrix were equal to one, in the neighbourhood of 10 diffusion instances. Considering a potentially strong block cipher with maximum diffusion capabilities, it is expected that each diffusion instance would include $(1/2) * n$ ones. Therefore, the ith diffusion instance would be expected to contribute with $(1/2)^i * n$ ones in the diffusion matrix. Alternatively, the probability that the calculated diffusion matrix for a potentially strong block cipher is not the actual one, would be $(1/2)^i$. It should also be highlighted that since the key information is not considered, the proposed approach is applicable only on block ciphers where their structure is not dependent on the key.

The diffusion matrix shows if a pairwise relation exists between input and output bits - that is, if a change of a particular input bit has the chance to affect a particular output bit. The diffusion matrix is very helpful in examining product ciphers, because it has the following property:

Lemma 1. *Let C be a FN of j rounds. The diffusion matrix of the cryptosystem is equal to:*

$$\mathcal{D}_C = \beta(\mathcal{D}_1 \cdot \mathcal{D}_2 \cdot \ldots \cdot \mathcal{D}_j) \tag{4}$$

where $\mathcal{D}_i$ is the diffusion matrix of the ith round and $\beta(\cdot) : N \rightarrow \{0, 1\}$ is defined as:

$$\beta(n) = \begin{cases} 1, & \textit{if } n \neq 0 \\ 0, & \textit{if } n = 0 \end{cases} . \tag{5}$$

Proof. The case of a two round FN is shown, that is $\mathcal{D} = \beta(\mathcal{D}_1 \cdot \mathcal{D}_2)$. Let $[\cdot]$ be a boolean evaluation, which evaluates the expression within the brackets to one if it is true and to zero is it is false, such as [p is prime]. The elements of $\mathcal{D}$, $\mathcal{D}_1$ and $\mathcal{D}_2$ are denoted by δ_{ij}, δ'_{ij} and δ''_{ij} respectively. Note that the output of round one is equal to the input of round two. For the first leftmost input bit it is:

$$[\text{input bit 1 is related with round-1 output bit } j] = \delta'_{1j},\ 1 \leq j \leq n \tag{6}$$

from the definition of the diffusion matrix. Similarly, for the first leftmost output bit:

$$[\text{output bit 1 is related with round-2 input bit } j] = \delta''_{j1},\ 1 \leq j \leq n\ . \tag{7}$$

Combining (6) and (7) we obtain:

$$[\text{input bit 1 is related with output bit 1}] = \delta'_{11} \cdot \delta''_{11} + \delta'_{12} \cdot \delta''_{21} + \ldots + \delta'_{1n} \cdot \delta''_{n1} \tag{8}$$

where the right-hand-side is a boolean expression, i.e. $. + .$ denotes the boolean **OR** and $. \cdot .$ denotes the boolean **AND**. If this is repeated for all input and output bits it gives:

$$[\text{input } i \text{ is related with output } j] = \delta_{ij} = \delta'_{i1} \cdot \delta''_{1j} + \delta'_{i2} \cdot \delta''_{2j} + \ldots + \delta'_{in} \cdot \delta''_{nj},\ 1 \leq i, j \leq n$$

or equivalently,

$$\mathcal{D} = \beta(\mathcal{D}_1 \cdot \mathcal{D}_2) \ . \qquad \square$$

From the diffusion matrix, we can calculate the diffusion, which is defined as the ratio of ones:

Definition 2. *The diffusion of a block cipher with a diffusion matrix $\mathcal{D}$ of size $(n \times n)$ is the quantity:*

$$D \triangleq \frac{\#\{\delta_{ij} | \delta_{ij} = 1, 1 \leq i, j \leq n\}}{n^2} \ . \tag{9}$$

Obviously, $D \in [0, 1]$. This definition of diffusion, combined with Lemma 1 can be used for assessing the diffusion of any product block cipher, provided that the diffusion matrices of the underlying rounds are known. We will demonstrate this by applying it onto FNs.

2.1 FN analysis

The diffusion matrix of a one round balanced FN would look like:

$$\mathcal{D} = \begin{bmatrix} \mathbf{O}_{n/2} & \mathbf{I}_{n/2} \\ \mathbf{I}_{n/2} & \mathbf{F} \end{bmatrix} \tag{10}$$

where $\mathbf{O}_{n/2}$ is a zero square submatrix, $\mathbf{I}_{n/2}$ is the identity submatrix and $\mathbf{F}$ is the diffusion matrix of the round function. In a balanced FN, all submatrices are of size $n/2$. The diffusion of this round would be equal to:

$$D_1 = \frac{4n + n^2 D_f}{4n^2} \tag{11}$$

where D_f is the diffusion of the round function. It can bee seen that the diffusion of a one round balanced FN is upper bounded by $(4 + n)/4n$ and therefore it cannot offer complete diffusion. To calculate the diffusion of a two round balanced FN, we apply Lemma 1:

$$\mathcal{D}_2 = \beta(\mathcal{D}_1 \cdot \mathcal{D}_1) = \begin{bmatrix} \mathbf{I}_{n/2} & \mathbf{F} \\ \mathbf{F} & \beta(\mathbf{F} \cdot \mathbf{F}) \end{bmatrix} \tag{12}$$

where it can be seen that the diffusion for a two round balanced FN can be at most $(3n^2 + 2n)/4n^2$. For a three round balanced FN, the diffusion can reach its maximum value, 1.

We observe that no matter how *strong* the round function is, the diffusion of a two round balanced FN is limited by the boundary 3/4. The reason for this is the structure of the diffusion matrix. The permutation of the columns of the matrix is directed by the Swapping Scheme, SS, which appears after the nonlinear transformation in a Feistel round. Although that the SS does influence the diffusion of the FN, it does not actually increase it; the increase is due to the application of the non-linear transformation. Typically, a SS is a permutation of the input bits. In a balanced FN the permutation is the swap between the $n/2$ leftmost bits and the $n/2$ rightmost bits. This swap is responsible for the symmetry in the diffusion matrix. However, each application of SS would not increase the diffusion:

Corollary 1. *The product encryption of a block cipher with diffusion equal to D and a SS, results to a cipher with the same diffusion (D).*

The proof follows from the fact that the diffusion matrix of the SS is a matrix with exactly n nonzero elements, arranged in a way that every row has exactly one nonzero element (i.e. the rank of the matrix is n). The identity SS is an instance of a SS where the diffusion matrix is the identity matrix.

The inherent structure of the FN diffusion matrix reveals the limitations of its diffusion capacity. Since the diffusion D measures the density of ones in the matrix, it follows that $1 - D$ would correspond to the density of zeros. It is therefore desirable that $1 - D$ reaches zero, in order to attain maximum diffusion. As observed above, in a two round FN with the ”traditional” swapping of the left and right input blocks, the number of zeros would be at least $1-(3n^2+2n)/4n^2$, i.e. it would reach asymptotically $1/4$ as n increases.

We now consider a two round Substitution Permutation Network, SPN [2], [12], where each round includes a non-linear function of the same diffusion D_1 as our FN above. For simplicity, it is assumed that these two rounds include different nonlinear functions, although their diffusion is the same, $D_1 = D_2$. We also consider the permutation to be a random SS, i.e. a random permutation of the input bits, rather than a tidy swapping of the left and right input block. The diffusion of the one round instances would be:

$$D_1 = D_2 = \frac{4n + n^2 D_f}{4n^2} \tag{13}$$

where D_f denotes the diffusion of the underlying nonlinear function. However, in a SPN construction it is possible that the zeros are placed randomly in the diffusion matrix. Therefore, the expected zeros in the diffusion matrix of the two round SPN for $D_f = 1$ would be (for the proof see Lemma 2, section 3):

$$(2(1 - D_1) - (1 - D_1)^2)^n = \left(\frac{15n^2 - 56n + 16}{16n^2}\right)^n \tag{14}$$

which is small (< 0.006) for most values of n ($n \geq 6$). From this result the inefficiency of FNs with respect to diffusion is apparent.

As mentioned above, Lemma 1 is useful when analysing the diffusion of product ciphers. For instance, FEAL-4 [13] is a four round FN with the characteristic that the leftmost half input is added (modulo 2) to the rightmost half input, before the first FN round. Considering the product encryption of the first addition and the first round, the diffusion at the end of the first round would be:

$$\beta(\begin{bmatrix} \mathbf{I}_{32} & \mathbf{O}_{32} \\ \mathbf{I}_{32} & \mathbf{I}_{32} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{O}_{32} & \mathbf{I}_{32} \\ \mathbf{I}_{32} & \mathbf{F} \end{bmatrix}) = \begin{bmatrix} \mathbf{O}_{32} & \mathbf{I}_{32} \\ \mathbf{I}_{32} & \mathbf{F} \end{bmatrix} \tag{15}$$

i.e. the additional complexity of the initial addition is completely redundant and unnecessary from a diffusion perspective, since for FEAL $D_f = 1$.

3 The diffusion randomness test

Statistical tests for randomness [14]-[16] are of a particular interest in cryptography, since they are one of the approaches for assessing the cryptographic strength of a cipher. This section describes a randomness test utilising the diffusion instances, Ψ.

For a potentially strong cipher, the number of zeros must be equal to the number of ones in every row of the diffusion instance. Furthermore, for a potentially strong cipher, (statistically) all runs of Ψ table constructions should result to having the number of ones equal to the number of zeros. However, such an examination does not give any indication about existing linear relations between the elements in the matrices. For instance, if $\psi_2[1] = \psi_3[2]$ with probability different to 0.5, there is a linear relation between input bits 1 and 2 [17].

The diffusion randomness test deals with the similarities of the diffusion instances, Ψ. For a potentially strong cipher the following criteria for the Ψ matrices are set:

- the number of ones should be equal to the number of zeros,
- the ones (and zeros) should be *randomly* distributed in the matrix,
- Ψ_i and Ψ_j should not be *similar* for $i \neq j$.

The first criterion denotes that the cipher is not biased toward ones or zeros. This is inherently related to the confusion of a cipher, where it is desirable that the chance of an output bit inverting is 0.5, given an inversion of an input bit. Published statistical tests for randomness, such as the frequency test [14] can be used.

The second and third criterion include arbitrary terms and need to be quantified. The test described in this paper attempts to provide means for measuring the randomness and similarity of the matrices as follows. The randomness test is based on the following Lemma.

Lemma 2. *Let* $\mathbf{A}$ *and* $\mathbf{B}$ *be two square matrices and* p_a *and* p_b *be the densities of zeros in each matrix respectively. If the zeros are distributed randomly in the matrices, then the* expected *density of zeros in their product* $\mathbf{C} = \mathbf{A} \times \mathbf{B}$ *would be:*

$$p_c = (p_a + p_b - p_a p_b)^n \tag{16}$$

where n *is the dimension of the matrices and the multiplication operation is performed in the set of integers.*

Proof. For $\mathbf{A}$, the density of zeros would be:

$$p_a = P(a_{ik} = 0) = \frac{\#(\text{zeros in } \mathbf{A})}{n^2} \tag{17}$$

Similarly, for B:

$$p_b = P(b_{kj} = 0) = \frac{\#(\text{zeros in } \mathbf{B})}{n^2} \; . \qquad (18)$$

For every element in C, the following relation holds:

$$c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj} \; . \qquad (19)$$

The probability to obtain a zero is obtained from (19):

$$P(c_{ij} = 0) = \prod_{k=1}^{n} P(a_{ik} = 0 \cup b_{kj} = 0) = (p_a + p_b - p_a p_b)^n \; . \qquad \square$$

By comparing the actual and estimated values, it is tested whether a cryptographic primitive behaves as a random source when generating the $\mathbf{\Psi}$ matrices. That is, in the case of a random source the zeros will be randomly placed in the matrices and there would be no consistent placement whatsoever. We argue that if the actual and estimated values are (statistically) different, then the underlying cryptographic primitive does not yield a pseudorandom function. The opposite is not necessarily true; a primitive passing the test does not imply that it is a pseudorandom function, since the test does not provide any indication about the computational indistinguishability of the primitive [18].

```
diff_rand_test(A,B){
  p_a = zeros_density(A);
  p_b = zeros_density(B);
  p_c = zeros_density(A*B);
  if (abs(p_c-(p_a+p_b-p_a*p_b)^n)>significance_level )
    then return ('fail')
    else return ('pass') }
```

Unfortunately for a relatively large n ($n > 40$) and $p_a, p_b < 2/3$, the density of zeros is negligible for both expected and actual values and therefore the randomness test would not produce significant results. Therefore it is suggested that the $\mathbf{\Psi}$ matrices are partitioned and the test is applied onto the partitions (submatrices). This is particularly applicable in FNs, where there are emerging submatrices due to the non uniformal treatment of input and output bits.

For the case of a balanced FN, the $\mathbf{\Psi}$ matrix would consist of four submatrices $\mathbf{Q}_i$ as follows:

$$\mathbf{\Psi} = \begin{bmatrix} \mathbf{Q}_1 & \mathbf{Q}_2 \\ \mathbf{Q}_3 & \mathbf{Q}_4 \end{bmatrix} \qquad (20)$$

and the test would then run as: diff_rand_test($\mathbf{Q}_i$,$\mathbf{Q}_j$), where $i \neq j$. It is expected that a three round balanced FN with an underlying round function being a pseudorandom function would pass the test, although that passing the test would not imply that the round function is pseudorandom. Applying this assumption to the well studied DES, it was established that the three round FN with the DES primitive did not pass the test, confirming the validity of the test (Table 1). The fact that DES could not pass the test is a direct consequence of the the inability of DES to reach complete diffusion in three rounds.

Table 1. Significant differences in DES

product	expected	actual	difference	diff_rand_test()
$\mathbf{Q}_1 \times \mathbf{Q}_2$	0.241739	0.216797	2.5	fail
$\mathbf{Q}_1 \times \mathbf{Q}_3$	0.204115	0.179688	2.4	fail
$\mathbf{Q}_1 \times \mathbf{Q}_4$	0.126188	0.077148	4.9	fail

4 Conclusions

Clearly the reason to adopt a FN structure in a block cipher is mainly due to the convenience it offers, such as ease of moving between encryption and decryption, and less due to its diffusion capabilities. High diffusion in a product cipher implies that the input bits are be treated uniformly in every round. Since this is not the case for a FN, additional complexity (e.g. more rounds) would be required. The proposed description and metric of diffusion enables both the investigation of the topology (structure) of a FN as well as the underlying non-linear function(s). This would allow the investigation of FNs consisting of different round functions, with varying input and output lengths as well as different swapping schemes (unbalanced heterogeneous FNs).

Although that the proposed approach initially aimed for studying FNs, most product block ciphers can benefit from such an analysis.

References

1. Feistel, H.: Block Cipher Cryptographic System, U.S. Patent #3,798,359 (1974).
2. Feistel, H., Notz, W. A., Smith, J. L.: Some Cryptographic Techniques for Machine-to-Machine Data Communications. Proceedings of the IEEE (1975) 1545–1554.
3. Schneier, B. and Kelsey, J.: Unbalanced Feistel networks and block cipher design. Proc. Fast Software Encryption, Lecture Notes in Computer Science, vol. 1039, Springer-Verlag (1996) 121–144.
4. Nakahara J. Jr., Vandewalle, J., Preneel, B.: Diffusion Analysis Of Feistel Networks (Extended Version). citeseer.nj.nec.com/article/nakahara99diffusion.html (1999).
5. Shannon, C. E.: Communication Theory of Secrecy Systems. Bell Systems Technical Journal, vol. 27 (1948) 623–656.
6. FIPS PUB 46: Data Encryption Standard. US Department of Commerce/ National Bureau of Standards (1977).
7. Goldreich, O., Goldwasser, S., Micali, S.: How to Construct Random Functions. Proceedings 25th Annual Symposium in Comp. Sci. (1984).
8. Luby, M. and Rackoff, C.: How to Construct Pseudorandom Permutations from Pseudorandom Functions. SIAM J. Computing, vol.17, no.2 (1988) 373–86.
9. Biham, E. and Shamir,A.: Differential cryptanalysis of DES-like cryptosystems. Journal of Cryptology. Vol. 4, No. 1 (1991) 3–72.
10. Webster, A. and Tavares, S.: On the design of S-boxes. In H. Williams (ed), Crypto’85, LNCS No. 218, Springer: Berlin Heidelberg New York (1986) 523–534.
11. Pfleeger, C.: Security in Computing. London: Prentice Hall (1989).
12. Heys, H. and Tavares, S.: Substitution Permutation Networks resistant to Differential and Linear cryptanalysis. Journal of Cryptology, no.9, vol. 1 (1996) 1–19.

13. Shimizu, A. and Miyaguchi, S.: Fast data encipherment algorithm FEAL. Advances in Cryptology, Eurocrypt'87, LNCS no.304, Springer: Berling Heidelberg New York (1988) 267–280.
14. Knuth, D.: Seminumerical algorithms. The Art of Computer Programming, vol 2. Addison-Wesley: New York (1981).
15. Rukhin, A., Soto, J., Nechvatal, V., Smid, M., Barker, E., Leigh, S. Levenson, M., Vangel, M., Banks, D., Heckert, A., Dray, J.: A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications. NIST Special Publication 800-22 (2000).
16. Beker, H. and Piper, F.: Cipher Systems: The Protection of Communications. Van Nostrand Reinhold (1982).
17. Matsui, M.: Linear Cryptanalysis Method for DES Cipher. Advances in Cryptology EUROCRYPT '93, LNCS 765 (1994) 386–397.
18. Blum, M. and Micali, S.: How to Generate Cryptographically Strong Sequences of Pseudo-Random Bits. SIAM Journal on Computing, Vol.13 (1984) 850–864.