A Novel Method of Constructing Block Ciphers Provably Resistant to Differential and Linear Cryptanalysis

Szymon Perlicki

April 2, 2022

Abstract

It is often the case that the size of some data is fixed and it might be encrypted all at once. Currently, in such a situation, we use a stream cipher or a block cipher with some mode of operation which has to be implemented additionally. Both of these options require storing additional information such as an IV, a nonce or a MAC, which might be a significant part of the data, if the data size is small. Unfortunately, up to now, ciphers with a larger block size have required a larger diffusion layer, which have taken up a lot of memory and made the cipher implementation harder. In this study a new method of constructing block ciphers is proposed. The presented construction consists of parallel SP-networks which recursively interchange data using a small diffusion layer, the size of which is recursively doubled by a presented algorithm. The method enables the creation of ciphers provably resistant to linear and differential cryptanalysis. These would be easy to parallelize and would make it possible to use a small, easy to store diffusion layer. The minimum required number of rounds for this method is derived. A proof is conducted, so that every encryption algorithm created using this method is resistant to linear and differential cryptanalysis under the given minimum required number of rounds.

1 Introduction

Diffusion layers are one of the most important parts of block ciphers in providing resistance to differential and linear cryptanalysis. S-boxes may have a very good differential uniformity and non-linearity, however, a cipher constructed by using them could still be broken using differential or linear cryptanalysis if it did not have a good enough diffusion layer. In SP networks a diffusion layer is applied to the whole block at once which is effective if the diffusion layer has a high branch number, but it is often the case that the diffusion layer is constructed of a few smaller ones, and even when those have high branch numbers, the resulting one can still have a low branch number which leads to using a

larger number of rounds in order to achieve a satisfying level of security. In this article, I will first present a way to create a $k2^{n+1}$ bit diffusion layer from a $2k$ bit diffusion layer with a certain property. This will be done in such a way that the created one will also have this property and therefore its size could be doubled using the same method and the process could be repeated indefinitely. In the next section, I will show how to use a diffusion layer with the property to create a cipher provably resistant to the linear and differential cryptanalysis and then I will prove its resistance under the given minimum required number of rounds.

2 Preliminaries

In order to be able to describe how to double a size of a diffusion layer, and how to use that diffusion layer to create a cipher provably resistant to linear and differential cryptanalysis, we first need to introduce a few definitions.

The notation:

- $+$ - the addition of binary vectors in $GF(2)$ or equivalently XOR of its components, it can also be the standard addition of numbers when it is clear from the context
- M^t - the transposition of the matrix M
- $\oplus$ - the direct sum of matrices
- $\cdot$ - the dot product of vectors
- $\mathbb{Z}_2 = \{0, 1\}$ - the set of all binary values
- $\mathbb{Z}_2^m$ - the set of all m-dimensional binary vectors
- the products of matrices are calculated in $GF(2)$
- $||$ - concatenation of vectors, namely

$$\begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_{k_1} \end{bmatrix} || \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_{k_2} \end{bmatrix} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_{k_1} \\ b_1 \\ b_2 \\ \vdots \\ b_{k_2} \end{bmatrix}$$

Definition 2.1. Two n-dimensional vectors a and b are *halves of a vector* v, if $v = a||b$.

Definition 2.2. Four n-dimensional vectors a, b, c, d are *quarters of a vector* v, if $v = a||b||c||d$.

Definition 2.3. Four $n \times n$ matrices A, B, C, D are *quarters of a matrix* E, if $E = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$.

Definition 2.4. A vector v is active, if $v \neq \vec{0}$.

Definition 2.5. Function f is a *diffusion layer*, if there exists a binary matrix A, such that $f(x) = Ax$.

Definition 2.6. Function $f : \mathbb{Z}_2^{2n} \mapsto \mathbb{Z}_2^{2n}$ activates both halves, if for every active n-dimensional vector v there exists a tuple of active n-dimensional vectors a, b, c, d such that $f(v||\vec{0}) = a||b$ and $f(\vec{0}||v) = c||d$, where $\vec{0}$ is an n-dimensional zero vector.

3 Doubling a size of a diffusion layer

Let M be an invertible binary matrix such that if a function $L : \mathbb{Z}_2^{2k} \mapsto \mathbb{Z}_2^{2k}$ is defined by the formula $L(x) = Mx$, then L activates both halves and let M_1, M_2, M_3, M_4 be the quarters of M, such that $M = \begin{bmatrix} M_1 & M_2 \\ M_3 & M_4 \end{bmatrix}$. At the input we have a $4k$-dimensional binary vector $v_0 = a||b||c||d$ and a diffusion layer L which takes only $2k$ bits, where a, b, c, d are quarters of v. Let's denote quarters of v_i by $Q_{i,1}, Q_{i,2}, Q_{i,3}, Q_{i,4}$ and do the following:

(1) Diffuse $Q_{0,1}$ and $Q_{0,2}$.
$v_1 = L(Q_{0,1}||Q_{0,2})||Q_{0,3}||Q_{0,4}$.

(2) Diffuse $Q_{1,3}$ and $Q_{1,4}$.
$v_2 = Q_{1,1}||Q_{1,2}||L(Q_{1,3}||Q_{1,4})$

(3) Diffuse $Q_{2,2}$ and $Q_{2,3}$.
$v_3 = Q_{2,1}||L(Q_{2,2}||Q_{2,3})||Q_{2,4}$

(4) Diffuse $Q_{3,1}$ and $Q_{3,4}$.
$v_4 = Q_{3,2}||Q_{3,3}||L(Q_{3,1}||Q_{3,4})$

(5) Restore the order of the quarters.
$v_5 = Q_{4,3}||Q_{4,1}||Q_{4,2}||Q_{4,4}$.

We can easily represent this transformation as a single diffusion layer $\bar{L}(v) = \bar{M}v$, where

$$\bar{M} = \begin{bmatrix} M_1 & 0 & 0 & M_2 \\ 0 & M_1 & M_2 & 0 \\ 0 & M_3 & M_4 & 0 \\ M_3 & 0 & 0 & M_4 \end{bmatrix} \begin{bmatrix} M & 0 \\ 0 & M \end{bmatrix}$$

The right factor of the product represents steps $1 - 2$ and the left, steps $3 - 5$.

This representation is not useful computationally but by using it, we can note a few facts about the diffusion as a single transformation, which will be useful, if we want to double the size of $\bar{M}$. First of all, we can see that $\bar{M}$ is invertible because all the described steps are invertible. Since steps $1-4$ involve only reversible diffusion L of two quarters and step 5 changes just the order of the quarters of v_4. Secondly, $\bar{L}$ maps $4k$ bits to $4k$ bits, so it is twice the size of L. The only thing left to prove is that $\bar{L}$ activates both halves.

Lemma 3.1. For every n-dimensional vector x and every $n \times n$ invertible matrix A, Ax is active $\leftrightarrow$ x is active.

Proof. Let's suppose that Ax is not active and x is active, it means that $Ax = \vec{0}$ and $x \neq \vec{0}$. Matrix A is invertible by the definition, therefore there exists A^{-1} such that $A^{-1}Ax = x$. But for every $2k \times 2k$ matrix B, we then have $BAx = B \cdot \vec{0} = \vec{0} \neq x$. Thus A is not invertible, which contradicts the assumption that A is invertible, thus, by contradiction, we know that if x is active, then Ax is active. There is also the case when x is not active, which means that $x = \vec{0}$ $\implies$ $Ax = A \cdot \vec{0} = \vec{0} \leftrightarrow Ax$ is not active. □

Theorem 3.2. *$\bar{L}$ activates both halves.*

Proof. It is enough to consider the case where only a single half of v_0 is active. In steps 1 and 2 there is no interaction between different halves, thus, from Lemma 3.1, v_2 has still only one active half. During steps 3 and 4, quarters from different halves are diffused and during step 5 their order is restored. There are two inactive quarters in the inactive half of v_2 and at least one active quarter in the active one, therefore the active quarter is diffused in step 3 or 4 along with one of the inactive quarters and since L activates both halves, there is at least one active quarter in each half of v_5. □

Since $\bar{M}$ is invertible and $\bar{L}$ activates both halves, $\bar{M}$ meets the requirements to be put in place of M and be doubled using the same method. The result of such doubling can be doubled again and so on.

Usually an s-box, not a vector, is called either differentially or linearly active but in this case it is a more useful definition because it all comes down to whether a difference between two vectors which is itself a vector, is non-zero and whether a mask which is also itself a vector, is non-zero. Therefore for a $4k$-dimensional binary vector x, a $4k$-dimensional input difference Δx and a $4k$-dimensional input mask Γx, we can examine the activity of the two halves of $\Delta y = \bar{L}(x) + \bar{L}(x + \Delta x) = \bar{L}(\Delta x)$ and the activity of the halves of $\Gamma y = \bar{M}^{-t}\Gamma x$, the relation between the input mask and the output mask has been proven in [2].

Since $\bar{L}$ activates both halves, if only one half of Δx is active, then both halves of Δy are active. There is also the case when both halves of Δx are active, then from Lemma 3.1 we know that Δy is active.

In order to examine the activity of Γy, it is useful to notice that, since $\bar{M}$ is invertible, $\bar{M}^{-t} = (\bar{M}^t)^{-1}$ is also invertible. We can deduce from this fact and Lemma 3.1, that if Γx is active, then Γy is active. Let's define the function $S : \mathbb{Z}_2^{4k} \mapsto \mathbb{Z}_2^{4k}$ with the formula $S(x) = \bar{M}^{-t}x$, which maps the input masks to the output masks. To prove that S activates both halves, we need a few additional lemmas.

Lemma 3.3. For every $n \times n$ matrix A, Ax is active for every n-dimensional active vector $x \leftrightarrow A$ is invertible.

Proof. We already know from Lemma 3.1 that if A is invertible, then Ax is active for every n-dimensional active vector x, so it suffices to prove that if A is not invertible, then there exists an active n-dimensional vector x such that Ax is not active. A is not invertible $\leftrightarrow$ function defined by the formula $f(x) = Ax$ is not a bijection $\leftrightarrow$ there exists a pair of n-dimensional vectors $x, y : x \neq y \wedge f(x) = f(y) \leftrightarrow Ax = Ay \leftrightarrow A(x-y) = \vec{0} \leftrightarrow A(x-y)$ is not active, but $x \neq y \leftrightarrow x - y \neq \vec{0} \leftrightarrow x - y$ is active. □

Lemma 3.4. For every $2n \times 2n$ matrix E, function $f : \mathbb{Z}_2^{2n} \mapsto \mathbb{Z}_2^{2n}$ defined by the formula $f(x) = Ex$ activates both halves $\leftrightarrow$ quarters A, B, C, D of E are all invertible.

Proof. Let v be an active $2n$-dimensional vector. We have to prove that if one half of an input to f is inactive, then both halves of the output are active.

$$f(v||\vec{0}) = E(v||\vec{0}) = \begin{bmatrix} A & B \\ C & D \end{bmatrix} (v||\vec{0}) = \begin{bmatrix} Av + B \cdot \vec{0} \\ Cv + D \cdot \vec{0} \end{bmatrix} = \begin{bmatrix} Av \\ Cv \end{bmatrix}.$$

As proven in Lemma 3.3, Av and Cv are active for all active n-dimensional vectors $v \leftrightarrow A$ and C are invertible.

From the definition of activating both halves, there is a second case to consider.

$$f(\vec{0}||v) = E(\vec{0}||v) = \begin{bmatrix} A & B \\ C & D \end{bmatrix} (\vec{0}||v) = \begin{bmatrix} A \cdot \vec{0} + Bv \\ C \cdot \vec{0} + Dv \end{bmatrix} = \begin{bmatrix} Bv \\ Dv \end{bmatrix}.$$

As proven in Lemma 3.3, Bv and Dv are active for all active n-dimensional vectors $v \leftrightarrow B$ and D are invertible. □

Lemma 3.5. If quarters A, B, C, D of an invertible matrix E are all invertible, then quarters of E^{-1} are all invertible.

Proof. From the partitioned matrix inversion formula [6], since A and D are invertible, we have

$$E^{-1} = \begin{bmatrix} A & B \\ C & D \end{bmatrix}^{-1} = \begin{bmatrix} (A - BD^{-1}C)^{-1} & -A^{-1}B(D - CA^{-1}B)^{-1} \\ -D^{-1}C(A - BD^{-1}C)^{-1} & (D - CA^{-1}B)^{-1} \end{bmatrix}$$

We can see that each quarter of E^{-1} is either an inversion of a matrix or a product of inversions and matrices invertible by the definition, therefore all of these quarters are invertible. □

Theorem 3.6. *For every $2n \times 2n$ matrix E, function defined by the formula $f(x) = Ex$ activates both halves $\leftrightarrow$ function defined by the formula $g(x) = E^{-t}x$ activates both halves.*

Proof. Let's represent E as a partitioned matrix $E = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$, where A, B, C, D are quarters of E. It follows that $E^{-t} = (E^t)^{-1} = \begin{bmatrix} A^t & C^t \\ B^t & D^t \end{bmatrix}^{-1}$. From Lemma 3.5 and the fact that A^t, B^t, C^t, D^t are all invertible, we know that quarters of $(E^t)^{-1} = E^{-t}$ are all invertible $\leftrightarrow$ from Lemma 3.4, g activates both halves. □

Theorem 3.7. *S activates both halves.*

Proof. From theorem 3.2, we know that $\bar{L}$ activates both halves and from theorem 3.6 it follows that S activates both halves. □

4 The method

In this section I will show how we can use a diffusion layer that activates both halves to create a cipher provably resistant to linear and differential cryptanalysis.

At the input there is a plaintext in the form of a $c2^n$-dimensional binary vector P and $r+1$ round keys[1] $K_1, K_2, \ldots, K_{r+1}$, each of them is $c2^n$-dimensional binary vector. During r rounds of encipherment, we want to transform the input data into a ciphertext consisting of $c2^n$ bits. Let's divide each of the 2^n c bit parts into l not necessarily equal parts of lengths $p_1, p_2, \ldots, p_l$ such that $\sum_{i=1}^{l} p_i = c$. To each of these parts, so to every $X \in \{1, 2, \ldots, 2^n\}$, we want to assign a sequence of bijective s-boxes $(S_{X,1}, S_{X,2}, \ldots, S_{X,l})$, such that $S_{X,i} : \mathbb{Z}_2^{p_i} \mapsto \mathbb{Z}_2^{p_i}$. Let M be a $c \times c$ invertible binary matrix such that the function defined by the formula $L(x) = Mx$ activates both halves.

Substitution of all c bits in the part numbered $X \in \{1, 2, \ldots, 2^n\}$ is defined as follows, $S_X(x) = S_{X,1}(x[1])||S_{X,2}(x[2])|| \ldots ||S_{X,l}(x[l])$, where $x[i]$ is the i-th part of x, of length p_i. The only thing left to define is an SDS layer (substitute-diffuse-substitute) of the whole state.

$$SDS(x_1||x_2|| \ldots ||x_{2^n}) = \\ = S_1(L(S_1(x_1)))||S_2(L(S_2(x_2)))|| \ldots ||S_{2^n}(L(S_{2^n}(x_{2^n})))$$

where for all $k : 1 \leq k \leq 2^n$, x_k is a c-dimensional vector.

[1] As in the other ciphers, the round keys will usually be derived from a single main key.

Let's denote L doubled k times as $L^{(k)}$ (e.g. $L^{(0)} = L$) and its matrix as $M^{(k)}$. Application of $L^{(k)}$ applied in parallel to the whole $c2^n$ bits state x is defined as (only for $k \leq n$)

$$D^{(k)}(x) = (\underbrace{M^{(k)} \oplus M^{(k)} \oplus \ldots \oplus M^{(k)}}_{2^{n-k}\ times})x$$

and the function which maps an input mask to an output mask is

$$S^{(k)}(x) = (\underbrace{M^{(k)} \oplus M^{(k)} \oplus \ldots \oplus M^{(k)}}_{2^{n-k}\ times})^{-t}x =$$
$$= (\underbrace{(M^{(k)})^{-t} \oplus (M^{(k)})^{-t} \oplus \ldots \oplus (M^{(k)})^{-t}}_{2^{n-k}\ times})x$$

We define

$$\eta_k = \begin{cases} SDS, & \text{for } k = 0 \\ \eta_{k-1} \circ D^{(k)} \circ \eta_{k-1}, & \text{for } 0 < k \leq n \end{cases}$$

The i-th round is then $R_i(x) = \eta_n(x + K_i)$ and the encryption $E(x) = (R_r \circ R_{r-1} \circ \ldots \circ R_1)(x) + K_{r+1}$. So the ciphertext is $E(P)$.

To define decryption, we need to define inversions of a few intermediate functions.

$$SDS^{-1}(x_1||x_2||\ldots||x_{2^n}) =$$
$$= S_1^{-1}(L^{-1}(S_1^{-1}(x_1)))||S_2^{-1}(L^{-1}(S_2^{-1}(x_2)))||\ldots||S_{2^n}^{-1}(L^{-1}(S_{2^n}^{-1}(x_{2^n})))$$

$$\eta_k^{-1} = \begin{cases} SDS^{-1}, & \text{for } k = 0 \\ \eta_{k-1}^{-1} \circ (D^{(k)})^{-1} \circ \eta_{k-1}^{-1}, & \text{for } 0 < k \leq n \end{cases}$$

$$R_i^{-1}(x) = \eta_n^{-1}(x + K_i)$$

and finally

$$E^{-1}(x) = (R_1^{-1} \circ R_2^{-1} \circ \ldots \circ R_r^{-1})(x + K_{r+1})$$

5 Proving resistance to linear and differential cryptanalysis

In order to prove resistance to linear and differential cryptanalysis of every cipher created using the method above, we need to first introduce a few definitions.

Definition 5.1. *Active s-box* is defined as an s-box given a non-zero input difference or a non-zero output mask value.

Note: When an s-box is bijective, the s-box given a non-zero output difference or a non-zero input mask value is also an active s-box. [5]

Definition 5.2. Number of active s-boxes in a c-dimensional binary vector x is defined as $W(x) = \#\{x[i] \mid i \in \mathbb{N} \wedge 1 \leq i \leq l \wedge x[i] \neq \vec{0}\}$.

Definition 5.3. *Branch number* of a $c \times c$ linear mapping D is given by $\mathcal{B}(D) = \min_{a \in Z_2^c \setminus \{\vec{0}\}} W(a) + W(D(a))$. [1]

Lemma 5.4. For every c-dimensional binary active vector x, c-dimensional linear mapping D and every $X \in \{1, 2, \ldots, 2^n\}$, $W(S_X(D(S_X(x))) + W(S_X(x)) \geq \mathcal{B}(D)$

Proof. Substitution does not change the number of active s-boxes because there is not any interaction between different s-boxes, thus $W(S_X(D(S_X(x))) = W(D(S_X(x)))$. Let $y = S_X(x)$, then, by the definition of $\mathcal{B}$, $W(S_X(D(S_X(x))) + W(S_X(x)) = W(D(y)) + W(y) \geq \mathcal{B}(D)$. □

Definition 5.5. *The number of active s-boxes during $\eta_0 = SDS$* is the sum of numbers of active s-boxes in the input mask or difference of the first and the second substitution.

From Lemma 5.4 we can see that the numbers of differentially and linearly active s-boxes during η_0 are greater than or equal to $\mathcal{B}(L)$ and $\mathcal{B}(S)$, respectively, where $S(x) = M^{-t}x$.

Definition 5.6. *The number of active s-boxes during $\eta_k = \eta_{k-1} \circ D^{(k)} \circ \eta_{k-1}$* is the sum of numbers of active s-boxes in the input mask or difference of the first and the second η_{k-1}, where k, $1 \leq k \leq n$, is an integer.

Definition 5.7. *The number of active s-boxes during a round* is equal to the number of active s-boxes during the corresponding η_n.

Lemma 5.8. For all $k \leq n$, there are at least $t_1 3^k \mathcal{B}(L)$ differentially active s-boxes and at least $t_2 3^k \mathcal{B}(S)$ linearly active s-boxes during η_k, where t_1 is the number of active $c2^k$ bit parts in an input difference of η_k and t_2 in its input mask.

Proof. We prove the lemma by induction. The application of η_0 consists of two substitutions, with 2^n parallel diffusions between them. Let's denote the number of active c bit parts in the input difference and mask of η_0 as t_1 and t_2, respectively. Those parts are substituted with no interaction between each other and diffused in parallel, so during the diffusion there is also no interaction between them. From Lemma 5.4 we know that during SDS of each active part, there are at least $\mathcal{B}(L)$ differentially active s-boxes and $\mathcal{B}(S)$ linearly active s-boxes and there are t_1 and t_2 such parts, thus there are at least $t_1\mathcal{B}(L) = t_1\mathcal{B}(L)3^0$ differentially active s-boxes and $t_2\mathcal{B}(L) = t_2\mathcal{B}(S)3^0$ linearly active s-boxes.

Let's suppose that the assertion is true for some k. Then consider an application of η_{k+1}, it consists of an application of η_k, a parallel diffusion of 2^{n-k-1} parts, each $2^{k+1}c$ bit and of a second application of η_k. The input difference

consists of t_1 active parts and the input mask of t_2 active parts, each part of $c2^{k+1}$ bits.

Let's consider all of these parts that have only one active half and denote the number of them as g_1 and g_2 in the input difference and mask of η_{k+1}, respectively. During the first η_k, there are at least $g_1 3^k \mathcal{B}(L)$ differentially active s-boxes and at least $g_2 3^k \mathcal{B}(L)$ linearly active s-boxes. After the parallel diffusions of $D^{(k+1)}$, since both it and $S^{(k+1)}$ consist of parallel $c2^{k+1}$ bit transformations that activate both halves, both halves of all of these parts in the output difference and mask of $D^{(k+1)}$ are active, the output difference and mask are input difference and mask of the second η_k, respectively. Therefore, there are at least $2g_1 3^k \mathcal{B}(L)$ differentially active s-boxes and at least $2g_2 3^k \mathcal{B}(S)$ linearly active s-boxes during the second η_k, that are jointly, during both of them, at least $g_1 3^k \mathcal{B}(L) + 2g_1 3^k \mathcal{B}(L) = g_1 3^{k+1} \mathcal{B}(L)$ differentially and at least $2g_2 3^k \mathcal{B}(S) + g_2 3^k \mathcal{B}(S) = g_2 3^{k+1} \mathcal{B}(S)$ linearly active s-boxes.

Now, let's consider all of these parts that have both halves active and denote the number of them as u_1 and u_2, in the input difference and mask of η_{k+1}, respectively. During the first η_k, there are at least $2u_1 3^k \mathcal{B}(L)$ differentially active s-boxes and at least $2u_2 3^k \mathcal{B}(S)$ linearly active s-boxes. After the parallel diffusions of $D^{(k+1)}$, from Lemma 3.1, we know that all of these parts remain active, thus during the second η_k, there are at least $u_1 3^k \mathcal{B}(L)$ differentially active s-boxes and at least $u_2 3^k \mathcal{B}(S)$ linearly active s-boxes, that is jointly, during both of them, at least $2u_1 3^k \mathcal{B}(L) + u_1 3^k \mathcal{B}(L) = u_1 3^{k+1} \mathcal{B}(L)$ differentially and $2u_2 3^k \mathcal{B}(S) + u_2 3^k \mathcal{B}(S) = u_2 3^{k+1} \mathcal{B}(S)$ linearly active s-boxes.

If we add up these two distinct cases, we have at least

$$g_1 3^{k+1} \mathcal{B}(L) + u_1 3^{k+1} \mathcal{B}(L) = t_1 3^{k+1} \mathcal{B}(L)$$

differentially active s-boxes and at least

$$g_2 3^{k+1} \mathcal{B}(S) + u_2 3^{k+1} \mathcal{B}(S) = t_2 3^{k+1} \mathcal{B}(S)$$

linearly active s-boxes.

By induction, the assertion is true for all natural numbers up to n, after which $D^{(x)}$ is undefined, for $x > n$. □

Theorem 5.9. *If an input difference and an input mask of a round are active, then during the round there are jointly at least $3^n \mathcal{B}(L)$ differentially active s-boxes and $3^n \mathcal{B}(S)$ linearly active s-boxes.*

Proof. We have $R_i(x) = \eta_n(x + K_i)$, adding a key does not change the number of active s-boxes in an input difference because the key cancels out. It also does not change the number of active s-boxes in an input mask because during the linear cryptanalysis, the key is considered fixed and it is implicitly absorbed into 0 in a linear approximation [4], so the input mask and the input difference of η_n are active. Thus, it follows immediately from definition 5.7 and Lemma 5.8 that during a round there are jointly at least $3^n \mathcal{B}(L)$ differentially active s-boxes and $3^n \mathcal{B}(S)$ linearly active s-boxes. □

Definition 5.10. For every input and output difference $\Delta x, \Delta y \in \mathbb{Z}_2^{p_i}$, the *propagation ratio* of each $S_{X,i}$ is defined as follows:

$$R_p^{S_{X,i}}(\Delta x \to \Delta y) = \frac{\#\{x \in \mathbb{Z}_2^{p_i} \mid S_{X,i}(x) + S_{X,i}(x + \Delta x) = \Delta y\}}{2^{p_i}}$$

Definition 5.11. For every input and output mask $\Gamma x, \Gamma y$, the *correlation coefficient* of each $S_{X,i}$ is defined as follows:

$$C_p^{S_{X,i}}(\Gamma x \to \Gamma y) = \frac{\#\{x \in \mathbb{Z}_2^{p_i} \mid x \cdot \Gamma x = S_{X,i}(x) \cdot \Gamma y\}}{2^{p_i - 1}} - 1$$

Definition 5.12. The maximal propagation ratio amongst all s-boxes is defined as

$$p = \max_{X,i,\Delta x \neq \vec{0}, \Delta y} R_p^{S_{X,i}}(\Delta x \to \Delta y)$$

Definition 5.13. The maximal correlation coefficient amongst all s-boxes is defined as

$$q = \max_{X,i,\Gamma x, \Gamma y \neq \vec{0}} C_p^{S_{X,i}}(\Gamma x \to \Gamma y)$$

It has been shown in [1] that the propagation ratio of a differential trail is equal to the product of propagation ratios of the active s-boxes assuming independence of the data entering s-boxes and it has also been shown there that the correlation coefficient of a linear trail is equal to the product of correlation coefficients of the active s-boxes assuming independence of the bits entering each s-box.

Theorem 5.14. *Assuming independence of the bits entering each s-box, the propagation ratio of every differential trail over r rounds of the cipher is upper bounded by $p^{3^n \mathcal{B}(L) r}$ and the correlation coefficient of every linear trail over r rounds is upper bounded by $q^{3^n \mathcal{B}(S) r}$.*

Proof. It follows from the facts that the propagation ratio of a differential trail can be represented as a product of $3^n \mathcal{B}(L) r$ propagation ratios, each upper bounded by p and that the correlation coefficient of a linear trail can be represented as a product of $3^n \mathcal{B}(S) r$ correlation coefficients, each upper bounded by q. □

For a $c2^n$ bit cipher to be resistant to differential cryptanalysis it is a necessary condition that there are no differential trails with the propagation ratio higher than 2^{1-c2^n} and to be resistant to linear cryptanalysis it is a necessary condition that there are no linear trails with correlation coefficient higher than $2^{-c2^{n-1}}$ over all but a few rounds (typically 2 or 3) [3]. Therefore, in order to derive the minimal required number of rounds, we have to solve two inequalities for r.

$$p^{3^n \mathcal{B}(L) r} \leq 2^{1-c2^n} \leftrightarrow r \geq \frac{1 - c2^n}{3^n \mathcal{B}(L)} \log_p(2)$$

$$q^{3^n \mathcal{B}(S) r} \leq 2^{-c2^{n-1}} \leftrightarrow r \geq \frac{-c2^{n-1}}{3^n \mathcal{B}(S)} \log_q(2)$$

Thus

$$r \geq \left\lceil \max\left(\frac{1 - c2^n}{3^n \mathcal{B}(L)} \log_p(2), \frac{-c2^{n-1}}{3^n \mathcal{B}(S)} \log_q(2) \right) \right\rceil$$

and it would be good to add a few rounds to this lower bound because we usually look for trails over less than all rounds and also to gain some security margin.

The assumption of independence in most cases will not be true but as noted in [1], it is close enough to the truth to provide a very good approximation of the propagation ratio and the correlation coefficient. It is also worth mentioning that the lack of existence of a single trail with the propagation ratio or the correlation coefficient higher than some value does not provide an assurance of a complete immunity to the differential and linear cryptanalysis but only of a resistance to some extent because different trails can combine to give jointly a significantly higher propagation ratio of a differential or a correlation coefficient of a linear approximation. For example, the proof of AES's immunity [3] to the linear and differential cryptanalysis is similar to this and it has only been proven there that there are no differential trails with the propagation ratio higher than 2^{-300} and that there are no linear trails with the correlation coefficient higher than 2^{-150} over 8 rounds of encryption and therefore the proof presented here should be considered correct on the same basis and with the same restrictions as the proof of AES's security.

6 Optional changes

There are a few things that might be changed in the cipher created this way, if one has reasons to do so, one might:

- Add a key after or before each diffusion during each round, as in SP-networks
- Replace η_0 with a customized SP-network instead of SDS. If during such an SP-network, there are at least t_1 differentially active s-boxes and at least t_2 linearly active s-boxes, it is easy to prove that during each round created with the method, there are at least $3^n t_1$ differentially active s-boxes and $3^n t_2$ linearly active s-boxes.

7 Arguments to use the presented method

There are a few reasons for which one might want to use the presented method:

- Being able to create a cipher with a large block size without using a large diffusion layer.

- Easy parallelization of a doubled diffusion layer, since steps 1 and 2 of the doubling can be run in parallel as well as steps 3 and 4.
- Easy parallelization of the encryption round, since both the substitution and the doubled diffusions are easy to run in parallel.

8 Warning

I would like to remind that a cipher created using the described method still has to be analysed in the context of the other attacks.

References

[1] J. Daemen. Cipher and hash function design strategies based on linear and differential cryptanalysis. 1995.

[2] J. Daemen, R. Govaerts, and J. Vandewalle. Correlation matrices. In *FSE*, 1994.

[3] Joan Daemen and Vincent Rijmen. Aes proposal: Rijndael, 1999.

[4] Howard Heys. A tutorial on linear and differential cryptanalysis. *Cryptologia*, 26, 06 2001.

[5] Masayuki Kanda, Youichi Takashima, Tsutomu Matsumoto, Kazumaro Aoki, and Kazuo Ohta. A strategy for constructing fast round functions with practical security against differential and linear cryptanalysis. In Stafford Tavares and Henk Meijer, editors, *Selected Areas in Cryptography*, pages 264–279, Berlin, Heidelberg, 1999. Springer Berlin Heidelberg.

[6] Tzon-Tzer Lu and Sheng-Hua Shiou. Inverses of 2 $\times$ 2 block matrices. *Computers and Mathematics With Applications - COMPUT MATH APPL*, 43:119–129, 01 2002.