Differential-ML Distinguisher: Machine Learning based Generic Extension for Differential Cryptanalysis

Tarun Yadav and Manoj Kumar

Scientific Analysis Group, DRDO, Metcalfe House Complex, Delhi-110 054, INDIA
{tarunyadav,manojkumar}@sag.drdo.in

Abstract. Differential cryptanalysis is an important technique to evaluate the security of block ciphers. There exists several generalisations of differential cryptanalysis and it is also used in combination with other cryptanalysis techniques to improve the attack complexity. Usefulness of Machine learning in differential cryptanalysis is introduced by Gohr in 2019 to attack the lightweight block cipher SPECK. In this paper, we present a framework to combine the classical differential distinguisher and machine learning (ML) based differential distinguisher. We propose a novel technique to construct differential-ML distinguisher which provides better results with reduced data complexity. This technique is demonstrated on lightweight block ciphers SPECK & SIMON where 96% & 99% (or more) success rate is achieved for distinguishing the 6-round SPECK and 9-round SIMON respectively.

Keywords: Block Cipher, Differential Cryptanalysis, Machine Learning

1 Introduction

Cryptanalysis of block ciphers witnessed remarkable progress after the proposal of differential attack on DES by Biham and Shamir [5] in 1990. Differential attack is the most basic and widely used cryptanalysis approach against block ciphers. This attack is generalised and combined with other cryptanalysis techniques to reduce the attack complexity. High probability differential trails are the first and foremost requirement for differential cryptanalysis to succeed. Matsui proposed a method to search the high probability differential trails based on branch-and-bound technique [12] in 1994. For large block sizes, classical approaches are not sufficient to provide the useful differential trails. In 2011, Mouha *et.al* proposed a new technique using mixed integer linear programming (MILP) to search the differential trails [13]. MILP based search method constructs the differential trails with better efficiency than branch-and-bound based methods.

Since, block ciphers are designed to thwart differential attack using wide trail design strategy [7] and Shannon's principles [10]. Therefore, existing trail search methods encounter a bottleneck for required data quickly and fail to provide the trails covering the required number of rounds. In practice, we need a differential

trail with probability greater than 2^{-n} to distinguish r rounds of an n-bit block cipher from random permutations. Whenever, the probability of an r-round trail becomes less than 2^{-n}, the trail is not useful for differential attack on r or more rounds of a block cipher. A differential trail is useful until it requires less data than available limit i.e. 2^n pairs. Therefore, the aim is to find a technique which can be used to extend the classical differential trails without increasing the data complexity. Machine learning based differential cryptanalysis approach works as a pretty solution to this problem.

In this paper, we combine the classical and machine learning techniques to design a ML based generic extension for any classical differential distinguisher. This provides the better results with a greater number of rounds and much lesser data complexity. We extend r-round high probability classical differential distinguisher ($D^{CD}_{1\cdots r}$) with s-round high accuracy ML based differential distinguisher ($D^{ML}_{r+1\cdots r+s}$) and combined distinguisher ($D^{CD\to ML}_{1..r+s}$) is used to distinguish $(r+s)$ rounds of a block cipher with much lesser data complexity. With this extension, the hybrid distinguisher outperforms both classical and ML based distinguisher. We experiment with two different types of lightweight block ciphers SPECK & SIMON and acquire the results with very high accuracy.

The remaining part of the paper is organised in the following manner. Section 2 discusses previous work related to ML based distinguisher. In Section 3, we provide the short description of lightweight block ciphers SIMON and SPECK. We discuss classical differential distinguisher and machine learning based differential distinguisher in section 4 and describe the existing work on differential distinguisher using machine learning in this section. In section 5, we propose our novel technique which combines the classical differential and ML based differential approaches. We demonstrate our technique on SPECK and SIMON block ciphers with high success rate and present the results of Differential-ML distinguisher in section 6. Finally, section 7 concludes the paper.

Conventions: Throughout this paper, we refer differential distinguisher with single input and output difference as a classical differential distinguisher.

2 Previous Work

Machine learning techniques are very helpful for big data analytics and it is used to determine minute relations in the data. In cryptology, identification of minute relations in the data plays an important role because these relations define the security strength of the cipher. In cryptanalysis domain, machine learning techniques for differential cryptanalysis are explored very recently and results are very promising.

Gohr[8] proposed the idea of learning differences for key recovery using machine learning. He presented a framework to construct the ML based differential distinguisher and used it for key recovery attack on SPECK32. Gohr compared this technique with classical differential attack and shown that data complexity for key recovery attack using ML distinguisher is less. Baksi et al.[1] used the same approach to design ML distinguisher for GIMLI cipher and GIMLI

hash[3]. Different ML architectures are compared in this work and claimed that ML distinguisher outperforms classical differential distinguisher. In comparison to Gohr's work[8], key recovery attacks are not demonstrated on GIMLI. In these previous work, ML based distinguishers have limitations on computation power, memory and data complexity. Due to these constraints, distinguisher cannot be extended beyond certain number of rounds and it becomes a major hindrance especially with a block size greater than 32.

3 SPECK and SIMON Block Ciphers

SPECK and SIMON are two families of block ciphers published by NSA[2] in 2013. These block ciphers are designed to provide high performance across a range of devices. There are 10 versions of each cipher based on the block and key size combinations which makes them suitable for wide range of applications. We discuss the encryption algorithm for 32-bit block size and 64-bit key versions of each block cipher. We omit the key expansion algorithm and NSA paper [2] can be refereed for more details. A brief description of SPECK and SIMON block cipher is provided in the following section.

3.1 Description of SPECK

SPECK32/64 is a block cipher with 32-bit block size and 64-bit key size. There are 22 rounds in SPECK32/64 block cipher. It is based on Feistel network and can be represented by composition of two Feistel maps. Encryption algorithm divides 32-bit input into two 16-bit words (X_{2i+1}, X_{2i}) and key expansion algorithm extract 16-bit round subkeys for each round. Round function comprises of addition modulo 2^{16}, bitwise XOR, left and right circular shift operations as described in Algorithm 1.

Algorithm 1: Encryption Algorithm of SPECK

Input: $P = (X_1||X_0)$ and K
Output: $C = (X_{65}||X_{64})$
for *i=1 to 22* **do**
  $X_{2i} = (X_{2i-1} \ggg 7 + X_{2i-2}) \oplus RK_{i-1}$
  $X_{2i+1} = (X_{2i-2} \lll 2 \oplus X_{2i})$
end

3.2 Description of SIMON

SIMON32/64 is a block cipher with 32-bit plaintext block and 64-bit secret master key. There are 32 rounds in SIMON32/64 block cipher and it is also based on Feistel network. Encryption algorithm divides the 32-bit input into two 16-bit words $X_{i+1}\&X_i$. Key expansion algorithm expands the 64-bit master key to provide 16-bit round subkeys (RK_i) for each round. It applies a round function

consisting bitwise XOR, bitwise AND, and left circular shift operations on left 16-bit words in each round as described in Algorithm 2.

Algorithm 2: Encryption Algorithm of SIMON

Input: $P = (X_1||X_0)$ and K
Output: $C = (X_{33}||X_{32})$
for *i=1 to 32* **do**
  $X_{i+1} = (X_i \lll 1 \& X_i \lll 8) \oplus (X_i \lll 2) \oplus X_{i-1} \oplus RK_{i-1}$
end

4 Differential Cryptanalysis

Differential attack is one of the most important analysis tool for cryptanalysis of block ciphers. This is the first attack of its own kind which reduced the complexity of DES better than exhaustive search [14]. Differential cryptanalysis created a path for several new cryptanalysis techniques like linear, impossible, algebraic and so on[6]. While designing an ideal block cipher, its output is tested for indistinguishability form random permutations. Although, there do not exists relationship between the single input and output occurrence of a block cipher, there may exist non-random relations in the input and output differences. The basic approach of differential attack is to study the propagation of input differences and exploitation of non-random relations between input and output differences. The classical differential attack works with a single differential trail providing the high probability relation between an input and output difference.

4.1 Classical Differential Distinguisher

A high probability differential trail is used for key recovery attack by adding some rounds on top/bottom of the trail. There exists several automated techniques to search the optimal differential trails for block ciphers[11] [9]. We extend 2-round SPECK and 4-round SIMON differential trail by ML based differential distinguisher. In this paper, we do not search for new differential trails for SIMON & SPECK but we use some part of existing differential trails published in [4].

4.1.1 Differential Trails for SPECK Biryukov *et.al*[4] presented the 9-round differential trails for Speck32/64 variant with probability 2^{-30}. We use 2-round differential trail with probability 2^{-6} in our experiment from 9-round trail presented in table 1.

Round Index	Input Difference $(\triangle X_{i+1}, \triangle X_i)$	Prob. $(-log_2 p_i)$
Δ_0	8054 A900	0
Δ_1	0000 A402	3
Δ_2	A402 3408	3
Δ_3	50C0 80E0	8
Δ_4	0181 0203	4
Δ_5	000C 0800	5
Δ_6	2000 0000	3
Δ_7	0040 0040	1
Δ_8	8040 8140	1
Δ_9	0040 0542	2

Table 1: Differential Trail of SPECK [4]

4.1.2 Differential Trails for SIMON Biryukov *et.al*[4] presented the 12-round differential trails for Simon32/64 variant with probability 2^{-34}. We use 5-round differential trail with probability 2^{-8} in our experiment from the 12-round trail presented in table 2.

Round Index	Input Difference $(\triangle X_{2i+1}, \triangle X_{2i})$	Prob. $(-log_2 p_i)$
Δ_0	0400 1900	0
Δ_1	0100 0400	2
Δ_2	0000 0100	2
Δ_3	0100 0000	0
Δ_4	0400 0100	2
Δ_5	1100 0400	2
Δ_6	4200 1100	4
Δ_7	1D01 4200	4
Δ_8	0500 1D01	8
Δ_9	0100 0500	3
Δ_{10}	0100 0100	2
Δ_{11}	0500 0100	2
Δ_{12}	1500 0500	3

Table 2: Differential Trail of SIMON [4]

4.2 ML Based Differential Distinguisher

For a chosen input difference, we use neural distinguisher design proposed by Gohr[8]. We also consider the improvements in this design suggested by Baksi et al.[1]. We use dense layers of MLPs (Multi Layers Perceptrons) instead of

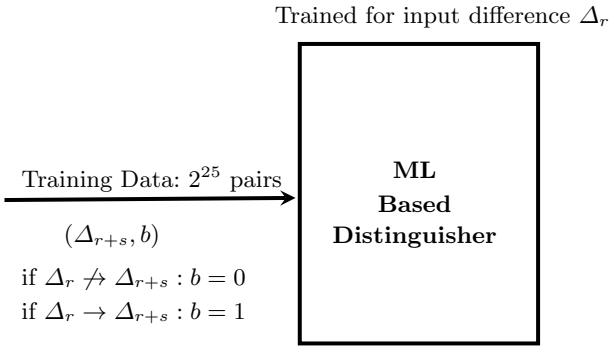

Fig. 1. Training Phase for ML Based Distinguisher

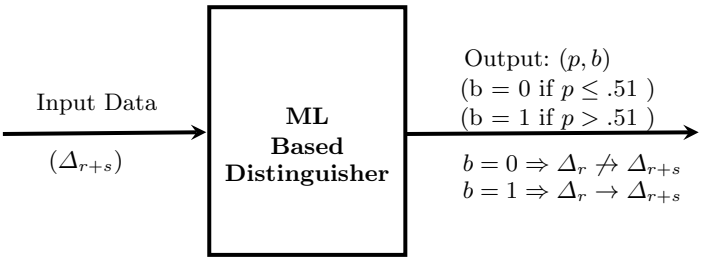

Fig. 2. Prediction using ML Based Distinguisher

convolution networks and trained the ML distinguisher on ciphertext differences rather than ciphertext pairs. These improvements make learning faster and efficient than Gohr's approach. Further, we use the same encryption key to generate the required training data because differential distinguisher is key independent. Therefore, we do not need to change the key for every encryption.

We train this ML distinguisher using the real and random differences approach proposed by Gohr. In this approach, half of the cipher text data belongs to the chosen plaintext difference and the other half belongs to the random plaintext differences. We label each ciphertext difference either with 1 if it belongs to the chosen input difference and label it with 0 if it does not belong to the chosen input difference. We provide this data to the MLP based model and train the model with 2 hidden layers having 1024 neurons each.

We assume that targeted system is accessible and there is no constraint on the training data required for learning the system. Therefore, we use 2^{25} ciphertext pairs for training phase. Out of these 2^{25} pairs, 2^{24} belongs to chosen plaintext difference and 2^{24} belongs to random plaintext differences as shown in Fig. 1. As described by Gohr, this approach works pretty well because not only specificity but also sensitivity is learned. Specificity and sensitivity are the learning of relations whether a ciphertext belongs to the chosen input difference or not respectively. We trained the model till the accuracy is saturated. This accuracy is the combination of accuracy of specificity and sensitivity. A model with accuracy greater than 0.51 (on the scale of 0 to 1) is considered as a distinguisher. After training phase, ML distinguisher will be able to distinguish any given ciphertext with a probability (p). We will label the ciphertext as 0 if probability is less than 0.51 and as 1 if probability greater than 0.51 as shown in Fig. 2. A distinguisher with higher accuracy will result in better prediction.

5 Differential-ML Distinguisher: an Extension for Classical Differential Distinguisher

Gohr's distinguisher lacks extendability because ML based distinguisher can only be designed if data requirement is computationally feasible. We propose a new approach to work with ML based distinguishers to overcome this constraint to a large extent. In our approach, we use ML based distinguisher in combination with classical differential distinguisher. ML distinguisher works as an extension to a classical distinguisher. We use an r-round classical differential trail and its output difference Δ_r is considered as an input difference for ML distinguisher. ML distinguisher is trained on this input difference (Δ_r) instead of plaintext difference (Δ_0). This new distinguisher reduces data complexity to a large extent with high accuracy.

To extend the r-round classical differential distinguisher, we consider the output difference Δ_r of the r-round classical differential trail and use Δ_r to train s-round distinguisher $D^{ML}_{r+1\cdots r+s}$. For training, half of the input data belongs to input difference Δ_r and half of the data belongs to random input differences. The ML based distinguisher is modelled with an accuracy α_i and we denote accuracy

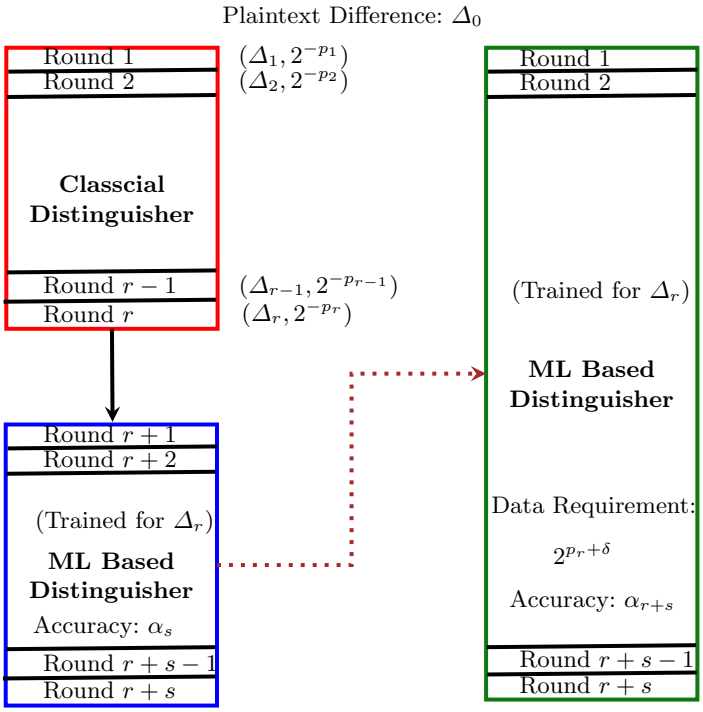

Fig. 3. Extension of Classical Distinguisher using ML Distinguisher

of s-round ML distinguisher as α_s. The accuracy α defines the strength of the distinguisher and better accuracy gives better predictions. Now, this distinguisher $D^{ML}_{r+1\cdots r+s}$ can distinguish any $(r+s)$-round ciphertext with high probability.

For r-round classical differential trail, output difference Δ_r with probability 2^{-p_r} requires 2^{p_r} data to get at least one occurrence of difference Δ_r. If we provide 2^{p_r} ciphertext pairs after $(r+s)$ rounds of encryption to $D^{ML}_{r+1\cdots r+s}$ then we expect $D^{ML}_{r+1\cdots r+s}$ to predict at least one occurrence of difference Δ_r. Although ML distinguisher works on multiple output differences, we expect it to learn the pattern of differences which are more frequent and suggested by the classical differential trail. Therefore, 2^{p_r} or more data is required for s-round distinguisher ($D^{ML}_{r+1\cdots r+s}$) to work as an $(r+s)$-round distinguisher ($D^{CD\rightarrow ML}_{1..r+s}$). Differential-ML distinguisher ($D^{CD\rightarrow ML}_{1..r+s}$) is a probabilistic distinguisher and we need to provide more data based on accuracy α_s for better prediction. Therefore, data complexity of $D^{CD\rightarrow ML}_{1..r+s}$ will be $2^{p_r+\delta}$, where δ defines the additional data required to make predictions with higher accuracy (Fig. 3).

In our experiments, we observe that $D^{CD\rightarrow ML}_{1..r+s}$ predicts ciphertexts belonging to the chosen plaintext difference Δ_0 with very high probability than random plaintext differences using $2^{p_r+\delta}$ data. We use this observation to increase the accuracy α_{r+s} by filtering higher probability predictions. We can always find a high probability threshold T and a cutoff C_T on the number of predictions with probability greater than T. With threshold T and cutoff C_T, we are able to achieve a very high success rate to distinguish $(r+s)$ rounds data. Experimental results in the next section show that data complexity for $(r+s)$ rounds using differential-ML distinguisher is far less than the classical differential distinguisher.

6 Experimental Results

We apply Differential-ML distinguisher on 32-bit variants of two light weight block ciphers SPECK and SIMON. We extend the classical differential distinguisher discussed in section 4 using 4-round Differential-ML distinguisher in each case. Using this technique, we have constructed 6-round Differential-ML distinguisher for SPECK and 9-round Differential-ML distinguisher for SIMON with very less data complexity than classical distinguisher.

6.1 Differential-ML distinguisher for SPECK

For SPECK32/64, we use the classical differential trail for initial 2 rounds as described in the table 1. Output difference after initial 2 rounds is 0xA4023408 (Δ_2) with probability 2^{-6}. We train ML distinguisher using Δ_2 as input difference for next 4 rounds.

6.1.1 Data Requirement

1. **Training:** Training data can be as large as possible because it does not contribute to the data complexity of the distinguisher. We have used 2^{25} ciphertext pairs for the training phase.

2. **Prediction:** Data used in predictions contributes to the data complexity of the distinguisher and it must be as small as possible. Differential probability for 2-round classical differential trail is 2^{-6}, therefore we require at least 2^6 data to get predictions for entire 6 rounds. To get higher accuracy, we require additional $2^{15}(\delta)$ data which increases the data complexity to 2^{21} for distinguishing 6-round SPECK.

6.1.2 Accuracy of Differential-ML Distinguisher (α_{r+s}) The 4-round ML distinguisher is trained with validation accuracy (α_s) 0.53. As described in section 5, it is used to extend 2-round classical distinguisher. The accuracy (α_{r+s}) of Differential-ML distinguisher for different experiments is mentioned in the table 3.

Experiment No.	Sample Size	Correctly Distinguished (True Positive, True Negative)
1	100	96(50,50)
2	100	96(50,50)
3	100	98(50,50)
4	100	99(50,50)
5	100	96(49,50)

Table 3: Accuracy for SPECK with $T = 0.60$ & $C_T = 4800$

In the experiments, 50 samples belong to the plaintext difference Δ_0(=0x8054 a900) of classical distinguisher and 50 samples belongs to random input differences. We use 2^{21} data and get 96% or more accuracy for each experiment. Therefore, data complexity of 6-round differential-ML distinguisher for SPECK is 2^{21}. However, data complexity of 6-round classical differential distinguisher is 2^{26} as mentioned in table 1. This shows that data complexity of 6-round differential-ML distinguisher is far better than classical differential distinguisher.

6.2 Differential-ML distinguisher for SIMON

For SIMON32/64, we use the classical differential trail for initial 5 rounds as described in the table 2. Output difference Δ_5 after 5 rounds is 0x11000400 with probability 2^{-8}. We use Δ_5 as input difference for training phase of 4-round ML distinguisher.

6.2.1 Data Requirement

1. **Training:** Training data requirements are similar to the case of SPECK. We use 2^{25} plaintext pairs for training the 4-round ML distinguisher.

2. **Prediction:** Similar to SPECK, we must reduce the data requirement for predictions. Differential probability for 5-round classical differential trail is 2^{-8}, therefore we require at least 2^8 data to distinguish 9-round SIMON. To increase the accuracy, 2^4 additional data (δ) is required. We require less additional data than SPECK due to high validation accuracy α_s of the ML distinguisher for SIMON. Due to this additional data, data complexity to distinguish 9-round SIMON is increased to 2^{12}.

6.2.2 Accuracy of Differential-ML Distinguisher (α_{r+s}) The 4-round ML distinguisher is trained with validation accuracy 0.98. It is used to extend 5-round classical differential distinguisher. The accuracy of 9-round Differential-ML distinguisher for different experiments is mentioned in the table 4.

Experiment No.	Sample Size	Correctly Distinguished (True Positive, True Negative)
1	100	100(50,50)
2	100	100(50,50)
3	100	100(50,50)
4	100	99(49,50)
5	100	100(50,50)

Table 4: Accuracy for SIMON with $T = 0.9997$ & $C_T = 5$

Similar to SPECK, 50 samples belong to the initial input difference Δ_0 (=0x04001900) of classical distinguisher and 50 samples belongs to random input differences. We use 2^{12} data to achive accuracy near to 100% for each experiment. Therefore, data complexity of 9-round differential-ML distinguisher is 2^{12}, while data complexity for 9-round classical differential distinguisher is 2^{27} (Table 2). This shows that differential-ML distinguisher provides much better results than classical differential distinguisher in case of SIMON also.

7 Conclusion

In this paper, we have proposed a novel technique to extend the classical differential attack using machine learning based distinguisher. Experimental results shows very high success rate for block ciphers SIMON & SPECK with a significant reduction in data complexity. We have also shown that we can extend any available classical differential distinguisher with machine learning based differential distinguisher. The new technique provides the better results in terms

of number of rounds and data complexity. This approach will open a new dimension for practical key recovery attacks using differential cryptanalysis where data complexity is a major roadblock.

References

1. Baksi, A., Breier, J., Dong, X., Yi, C.: Machine Learning Assisted Differential Distinguishers For Lightweight Ciphers. https://eprint.iacr.org/2020/571, (2020)
2. Beaulieu, R., Shors, D., Smith, J., Treatman-Clark, S., Weeks, B., Wingers, L.: The SIMON and SPECK families of lightweight block ciphers. Cryptology ePrint Archive, Report 2013/404. https://eprint.iacr.org/2013/404, (2013)
3. Bernstein, D.J., Kolbl, S., Lucks, S., Massolino, P.M.C., Mendel, F., Nawaz, K., Schneider, T., Schwabe, P., Standaert, F., Todo, Y., Viguier, B.: Gimli, (2019)
4. Biryukov, A., Roy, A., Velichkov, V.: Differential Analysis of Block Ciphers SIMON and SPECK. FSE 2014, 546–570, LNCS, Volume 8540, (2014)
5. Biham, E., Shamir, A.: Differential Cryptanalysis of the full 16-round DES, CRYPTO 92, LNCS, Vol. 740, 487–496, Springer, (1992)
6. Bogdanov, A.: Analysis and Design of Block Cipher Constructions, Ph.D. thesis, (2009)
7. Daemen, J., Rijmen, V.: The Design of Rijndael, Springer-Verlag, (2002)
8. Gohr, A.: Improving attacks on round-reduced SPECK32/64 using deep learning. In Boldyreva, A., Micciancio, D., eds.: Advances in Cryptology- CRYPTO 2019, Cham, Springer International Publishing, 150–179, (2019)
9. Hays, H.M.: A Tutorial on Linear and Differential Cryptanalysis, Cryptologia, Vol. 26, No. 3, 188–221, (2002)
10. Knudsen, L., Robshaw, M.J.B.: Block Cipher Companion, Book Springer, ISBN 978-3-642-17341-7, (2011)
11. Kumar, M., Suresh, TS, Pal, S.K., Panigrahi, A.: Optimal Differential Trails in Lightweight Blokc Ciphers ANU and PICO, Cryptologia, Vol. 44, No. 1, 68–78, (2020)
12. Matsui, M.: On Correlation between the Order of S-boxes and the Strength of DES, EUROCRYPT 94, LNCS, Vol 950, 366–375, Springer, (1994)
13. Mouha, N., Wang, Q., Gu, D., Preneel, B.: Differential and linear cryptanalysis using mixed-integer linear programming. In: Information Security and Cryptology - 7th International Conference, Inscrypt 2011, Beijing, China, November 30 - December 3, 2011. Revised Selected Papers. 57–76, (2011)
14. US National Bureau of Standards, Data Encryption Standard. Federal Information Processing Standards Publications, vol. 46, (1977)