KEY-DEPENDENT APPROXIMATIONS IN CRYPTANALYSIS.

AN APPLICATION OF MULTIPLE $\mathbb{Z}_4$ AND NON-LINEAR APPROXIMATIONS.

FX Standaert, G Rouvroy, G Piret, JJ Quisquater, JD Legat

Universite Catholique de Louvain, UCL Crypto Group,
Place du Levant, 3, 1348 Louvain-la-Neuve,
standaert,rouvroy,piret,quisquater,legat@dice.ucl.ac.be

Linear cryptanalysis is a powerful cryptanalytic technique that makes use of a linear approximation over some rounds of a cipher, combined with one (or two) round(s) of key guess. This key guess is usually performed by a partial decryption over every possible key. In this paper, we investigate a particular class of non-linear boolean functions that allows to mount key-dependent approximations of s-boxes. Replacing the classical key guess by these key-dependent approximations allows to quickly distinguish a set of keys including the correct one. By combining different relations, we can make up a system of equations whose solution is the correct key. The resulting attack allows larger flexibility and improves the success rate in some contexts. We apply it to the block cipher Q. In parallel, we propose a chosen-plaintext attack against Q that reduces the required number of plaintext-ciphertext pairs from 2^{97} *to* 2^{87}.

1. INTRODUCTION

In its basic version, linear cryptanalysis is a known-plaintext attack that uses a linear relation between input-bits, output-bits and key-bits of an encryption algorithm that holds with a certain probability. If enough plaintext-ciphertext pairs are provided, this approximation can be used to assign probabilities to the possible keys and to locate the most probable one.
Non-linear approximations are commonly used in cryptanalysis in order to increase the probabilities of linear approximations. The key-dependency of non-linear approximations is mentioned in several papers and allows more flexibility in the attacks. In this paper, we underline how a particular class of non-linear approximations can be used to mount interesting key-dependent relations. Then, we combine these relations to make up a system of equations whose solution is the secret key. It allows a great flexibility in the number of key-bits we want to recover and improves the success rate in some contexts. We apply the method to the block cipher Q, where potential improvements exist in multiple approximations. We also improve the linear cryptanalysis of Q by moving to the chosen-plaintext context.
This paper is organized as follows. Section 2 lists some previous works in the area of cryptanalysis that are closely related to this paper. Section 3 explains the basic principle of our attack. Section 4 describes the block cipher Q as well as an iterative linear characteristic of the block cipher. We stress that moving to the chosen-plaintext context can significantly improve the linear attack. Section 5 compares our key-dependent attack with the classical attack against Q. Finally, section 6 poses some open problems and underlines potential improvements. Conclusions are in section 7.

* This work has been funded by the Walloon region (Belgium) through the research project TACTILS.

2. RELATED WORKS

Since the first description of linear cryptanalysis by Matsui [9], plenty of papers tried to take advantage of the method in different attempts to break public ciphers and some of these papers include theoretical improvements. We try here to list the improvements that are directly connected to our work.
In 94, Nyberg [10] introduced the concept of linear hull and explains why the practical success rate of the attack is sometimes better than theoretically predicted by Matsui. In the same time, Kaliski and Robshaw [11] investigate the possibility to use multiple linear approximations in order to improve this success rate. Although practical improvements in the cryptanalysis of DES are limited, their work underlines situations where multiple approximations can be efficiently combined.
In 96, Knudsen and Robshaw [12] tried take advantage of non-linear approximations to improve the probabilities of the characteristics used to approximate block ciphers. They underlined the key-dependency of non-linear approximations and concluded that the problem of connecting them together is complex. Practically, their improvements are limited to the outer rounds of block ciphers. Shimoyama and Kaneko use similar ideas to improve the cryptanalysis of DES in 98 [13].
In 2000, Knudsen and Mathiassen [14] illustrate that moving from the known plaintext to the chosen plaintext context can improve matters in linear cryptanalysis. Finally, we refer to the recent work of Parker and Raadum[1] [7, 8], who generalize linear cryptanalysis to larger fields than $\mathbb{Z}_2$.
This paper tries to take advantage of $\mathbb{Z}_4$ approximations when we directly recombine them into the binary case. This allows to get several approximations with very interesting biases. Then, we investigate the possibility to mount key-dependent relationships and underline that these can help to quickly distinguish a small set of keys including the correct one. By combining different approximations, we can find a system of equations whose solution is the correct key. Potential advantages could then be found in the large number of equations making up the system.
In terms of connections, these results are directly connected to $\mathbb{Z}_4$ cryptanalysis but the recombination into the binary case can be viewed as a particular case of non-linear approximations. However, our objective is to underline that the classical way to perform the key guess, using a real s-box in the last round, is not necessarily the best way to do it. Key-dependency in non-linear approximations offers interesting alternatives. The combination of several equations is also closely related to multiple approximations. Finally, we take advantage of the chosen-plaintext context in order to improve the cryptanalysis of the block cipher Q.

3. BASIC PRINCIPLE

We assume that the reader is familiar with linear cryptanalysis as well as with its improvements presented in section 2.

3.1 $\mathbb{Z}_4$ Approximations of s-boxes: Let's take a simple 4-bit×4-bit substitution

[1]Technical reports of the NESSIE project.

box. For example:

$$SB = \{0, 15, 11, 8, 12, 9, 6, 3, 13, 1, 2, 4, 10, 7, 5, 14\} \quad (1)$$

In a linear attack, we try to approximate this s-box with a linear boolean function. For every output bit, there exist 2^4 different linear functions and if we combine output bits together, we have $2^4 \times 2^4$ possible linear approximations of the s-box. The problem of finding good linear approximations is easily done by exhaustive search. Practically, the best linear approximations of SB holds with a bias $\varepsilon = 4/16$ [2].
Obviously, other approximations are possible. For example, non-linear approximations offer more possibilities by combining XOR (or addition modulo 2) operations with AND (or multiplication modulo 2) operations. The classical problem in non-linear cryptanalysis is to combine these non-linear relations with key addition layers. In linear cryptanalysis, key additions only influence the sign of the bias. When using non-linear approximations, also the value of the bias is key-dependant.
In [7, 8], a generalization of the linear cryptanalysis in $\mathbb{Z}_4$ is proposed. We observed that a simple recombination of this generalization into the binary case allows to observe very interesting biases. In this paper, we propose to approximate n-input s-boxes with functions $f : (\mathbb{Z}_4)^n \rightarrow \mathbb{Z}_2$ whose coefficients are in $\mathbb{Z}_4$. We recombine symbols (or function outputs) of $\mathbb{Z}_4$ into $\mathbb{Z}_2$ with the simple rule:

1. Symbols 0,1 in $\mathbb{Z}_4$ are 0 in $\mathbb{Z}_2$.
2. Symbols 2,3 in $\mathbb{Z}_4$ are 1 in $\mathbb{Z}_2$.

The obvious consequence of this generalization is a larger number of possible approximations, moving from 2^n in the binary case to 4^n in $\mathbb{Z}_4$. However, this is nothing else than a kind of non-linear cryptanalysis. Indeed, when we recombine into $\mathbb{Z}_2$, coefficients 1 and 3 will give rise to quadratic terms and coefficients 2 and 3 to linear terms. For example:

$$y_1 = (3x_0 + 2x_1 + x_2 + x_3) mod_4 \quad (3)$$

in $\mathbb{Z}_4$ becomes[3]

$$y_1 = x_0 \oplus x_1 \oplus x_0x_2 \oplus x_0x_3 \oplus x_2x_3 \quad (4)$$

when we recombine[4] it into $\mathbb{Z}_2$ and it holds with a bias $\varepsilon = 6/16$.

3.2 Key Dependent Approximations: Now the question remains: "What can we do with these non-linear relations?". As the problem of combining non-linear approximations together has not changed, we propose here to take advantage of the

[2]In this paper, we define the bias of a linear or $\mathbb{Z}_4$ approximation that holds with probability p as $\varepsilon = p - 1/2$. This definition allows to determine the probability P of an approximation involving several active s-boxes approximated with biases ε_i (pilling-up lemma):

$$P = \frac{1}{2} + 2^{n-1} \prod_{i=1}^{n} \varepsilon_i \quad (2)$$

[3]We mean that both functions have the same truth table.

[4]$3x_0$ and $2x_1$ give rise to linear terms x_0 and x_1. Then we have 3 quadratic terms x_0x_2, x_0x_3 and x_2x_3 (because x_2 and x_3 have coefficient 1 and x_0 has coefficient 3).

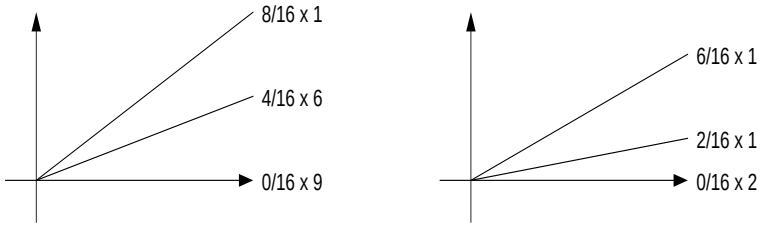

Figure 1: Linear Cryptanalysis and Key Dependant Attack.

key-dependant behavior of these relations. We can easily investigate the influence of this key addition on a non-linear approximation. for example, if the 4-bit input of the s-box is XORed with 4 key-bits, we have:

$$z_1 = 3.(x_3 \oplus k_3) + (x_2 \oplus k_2) + 2.(x_1 \oplus k_1) + 3.(x_0 \oplus k_0) \tag{5}$$

The next table gives the probability that 5 holds depending on the key added:

K	Probability		K	Probability		K	Probability		K	Probability
0	14/16		4	6/16		8	8/16		12	8/16
1	8/16		5	8/16		9	6/16		13	14/16
2	2/16		6	10/16		10	8/16		14	8/16
3	8/16		7	8/16		11	10/16		15	2/16

We observe that the largest bias ($\varepsilon = 6/16$) can happen for 4 different keys (0,2,13,15). Remark that a coefficient 2 only introduce linear terms in the boolean function and therefore, the associated key bits do not affect the magnitude of the bias. For example, the bias magnitude of (4) is independent of bit k_1 which cause $K = 0$ and $K = 2$ to have the same bias magnitude.

Definition: Let equation Y be considered "independent" of equation X if the highest bias achieved by X suggest a set of k (in our example $k = 4$) keys, and the highest bias achieved by Y suggest another set of keys, so that X and Y do not suggest the same keys.

Definition: Let a complete set of equations be a set of equations where every one of the 2^n possible keys are suggested by only one equation.

A simple exhaustive search allowed us to find several complete sets of equations. For a complete set of equations of SB, the biases are always distributed in the following way:

$$\begin{aligned} \varepsilon(good.case) &= \pm 6/16, \quad & \varepsilon(bad.case_1) &= \pm 0/16 \\ \varepsilon(bad.case_2) &= \pm 0/16, \quad & \varepsilon(bad.case_3) &= \pm 2/16 \end{aligned} \tag{6}$$

Distinguishing the good case obviously involves determining that the key is in a 4-element set.

3.3 An Attack using Key Dependent Approximations: In a classical linear cryptanalysis against a r-round block cipher, the attacker uses a $(r-1)$-round

approximation. Then the ciphertext is partially decrypted through the final round under every possible key in order to get the active bits of the linear approximations at round $r-1$. The linear approximation is checked for every possible key and the correct key will cause the relation to hold more significantly. However, some wrong keys will also cause the approximation to hold, but with a lower bias, as shown in Figure 1. In our practical example with SB, the problem is to distinguish a curve (representing the good key) that holds with a bias $\varepsilon = \pm 8/16$ from 6 curves with bias $\varepsilon = \pm 4/16$ and 9 with bias $\varepsilon = 0/16$. The resulting success rate depends on the probability that the linear approximation holds, the number of plaintext-ciphertext pairs and the correlation between the different curves.

Computation of the Success Rate [9]: Let N be the number of given random plaintext-ciphertext pairs and p be the probability that the linear approximation holds (assume $|p-\frac{1}{2}|$ is sufficiently small). Let $q^{(i)}$ be the probability that a wrong key candidate $K_w^{(i)}$ produces the same partial decryption in the final round than the correct key K_g. Then, if $q^{(i)}$'s are independent, the success rate of the attack is:

$$\int_{x=-2\sqrt{N}|p-\frac{1}{2}|}^{\infty} \Big(\prod_{K_w^{(i)} \neq K_g} \int_{-x-4\sqrt{N}(p-\frac{1}{2})q^{(i)}}^{x+4\sqrt{N}(p-\frac{1}{2})(1-q^{(i)})} \frac{1}{\sqrt{2\Pi}} e^{\frac{-y^2}{2}} dy \Big) \frac{1}{\sqrt{2\Pi}} e^{\frac{-x^2}{2}} dx \qquad (7)$$

In our alternative approach, the final round is replaced by a key-dependent $\mathbb{Z}_4$ approximation. Then we decrypt the ciphertext under the 4 equations of a complete set and one of these equations will cause the linear approximation to hold more significantly. In our practical example with SB, the problem is to distinguish a curve (representing the good 4-element set of keys) that holds with a bias $\varepsilon = \pm 6/16$ from 1 curve with bias $\varepsilon = \pm 2/16$ and 2 with bias $\varepsilon = 0/16$. In section 5, we apply both approaches to the block cipher Q and compare them. We also investigate how different sets of equations can be combined to recover the key.

4. THE BLOCK CIPHER Q

4.1 Description: Q [3] is a block cipher submitted as a candidate to the NESSIE project. It is already broken by differential and linear cryptanalysis [4, 5]. It has a straightforward SPN structure with s-boxes based on those in Rijndael (The AES selection) and Serpent, leaving out linear transformations excepted a simple permutation of the bytes. As a result, its diffusion properties are suboptimal. Q has 128-bit text and key blocks. The block is divided into 4 words and 16 bytes as shown in Figure 2. The round function of Q is represented in Figure 3 and is repeated 8 or 9 times in order to get a secure cipher. In the round function, Bytesub is taken from Rijndael. It substitutes the value of each byte independently. The bit-slice s-boxes (SA and SB) are taken from Serpent. For $i = 0, ..., 31$, we construct a 4-bit input i by taking bit i from every word, then we replace each input according to the s-box and return the new bit values to their original place. Finally, the permutation changes the order of the bytes in the words in the following way: word 0 is not changed, word 1 is rotated by 1 byte: (4,5,6,7) becomes (7,4,5,6), word 2 is rotated by 2 bytes and word 3 is rotated by 3 bytes. All these transforms are combined with classical key addition

3	7	11	15
2	6	10	14
1	5	9	13
0	4	8	12
W0	W1	W2	W3

Figure 2: Q Blocks.

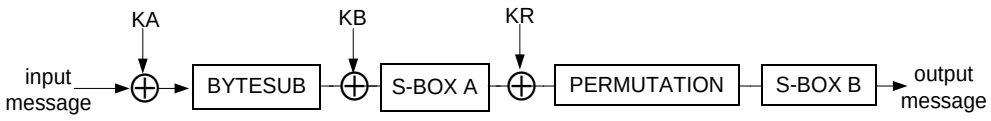

Figure 3: Q Round.

layers (KA, KB, KR). For clarity, we ignore the modifications of the first and final rounds. Only a final key addition is added after the last s-box SB[5].

4.2 A Simple and Iterative Linear Approximation: Due to the very simple structure of Q, movements of bits and bytes are specific for every part of the round:

1. A vertical shift of bytes is achieved by the permutation only.
2. An horizontal shift of bytes is achieved by s-boxes SA and SB only.
3. Bit modifications inside one byte are achieved by Bytesub only.

Combining this observation with the absence of a specific diffusion layer, we can easily build an iterative characteristic with only one active s-box in every layer of the cipher. Our iterative characteristic uses the following approximations and we illustrate the first round in Figure 4 where the grey bytes are active.

Bytesub:

1. Round 1 : $x_7 \oplus x_0 = y_7$, $\varepsilon = 16/256$.
2. Round 2 : $x_7 = y_2$, $\varepsilon = 16/256$.
3. Round 3 : $x_2 = y_5$, $\varepsilon = 16/256$.
4. Round 4 : $x_5 = y_7$, $\varepsilon = 12/256$.

SA **and** SB**: (the same approximations are used in every round)**

1. SA: $x_0 = y_1$, $\varepsilon = 2/16$.
2. SB: $x_1 = y_0$, $\varepsilon = 2/16$.

The output of round 4 can be linked to the input of round 2 if we want more rounds to be approximated. On the same figure, we illustrate that by moving to a chosen plaintext context, we can easily fix the input bits of Bytesub and SA in the first round (black bytes are fixed). This provides a significant improvement of the probability that the linear approximation holds by a factor 2^5. Equations 8, 9, give the probabilities that our 4-round approximation holds in a known- or chosen-plaintext context.

$$P_{known.plaintext} = 2^{11}.(\frac{1}{16}.\frac{1}{8}.\frac{1}{8})^3.(\frac{12}{256}.\frac{1}{8}.\frac{1}{8}) = 2^{-29.41} \quad (8)$$

$$P_{chosen.plaintext} = 2^{9}.(\frac{1}{16}.\frac{1}{8}.\frac{1}{8})^3.16.8.(\frac{12}{256}.\frac{1}{8}.\frac{1}{8}) = 2^{-24.41} \quad (9)$$

[5]The bias ε is defined in the appendix.

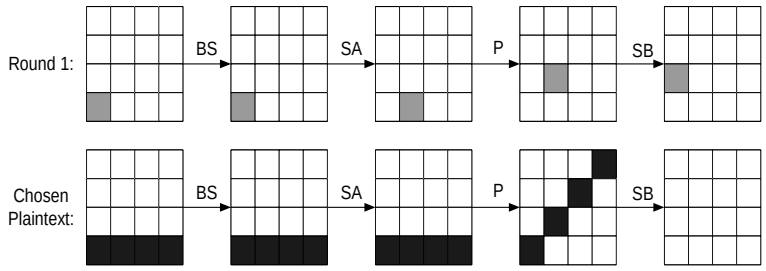

Figure 4: Iterative characteristic and chosen plaintext effect.

Note that applying this simple observation to the best-known attack against 8 rounds of Q [5], we can improve the probability of the approximation by a factor 2^5 and the number of plaintext needed by about 2^{10}. The resulting attack would need about 2^{87} chosen-plaintexts to reach a success rate of 98.4%. The 32 fixed bits leave us with 2^{96} chosen-plaintexts to mount an attack and therefore do not restrict the plaintext area too much.

Finally, it is important to note that by changing approximations of SA and SB (with the same biases), we can easily construct similar characteristics where we change the last active s-box SB[6]. Every one of these characteristics allow to recover 4 different key bits with the same plaintext-ciphertext set. For example,by simply changing the approximation of SA in the final round, ($x_0 = y_1$, $x_0 = y_2$ and $x_0 = y_3$ have the same bias), we could have bytes 4, 9 and 14 active in the last round and recover 12 key bits. More approximations exist if more bits are needed. Every attempt to recover 4 key bits is an independent experiment and if p is the probability to recover 4 key bits, the probability to recover $4 \times n$ key bits equals the probability that the n attempts success: $P = p^n$.

5. A CHOSEN-PLAINTEXT ATTACK AGAINST 4 ROUNDS OF Q

5.1 Comparison: In this section, we evaluate the success rate of our attack against 4 rounds of Q and compare it to a classical key guess.
In the classical case, we have 9 active s-boxes because the first Bytesub and SA are fixed and the last SB is used to perform the key guess. The resulting probability is:

$$P_{chosen.plaintext} = 2^8.\frac{1}{8}.(\frac{1}{16}.\frac{1}{8}.\frac{1}{8})^2.(\frac{12}{256}.\frac{1}{8}) = 2^{-22.41} \tag{10}$$

As mentioned in section 3, the keys are correlated in the following way: the correct key exhibits the best bias ($\pm 8/16$), 6 wrong keys exhibit a bias $\pm 4/16$ and the 9 last ones have a bias $\pm 0/16$.

When using a key-dependent approximation with bias $\varepsilon = \pm 6/16$ in the last round, we have 10 active s-boxes because only the first Bytesub and SA are fixed. The resulting

[6]The same reason causes a significant linear hull effect.

Method/N	2^{42}	2^{43}	2^{44}	2^{45}	2^{46}	2^{47}	2^{48}	2^{49}
Classical	11.17	16.36	25.08	38.62	56.71	76.57	92.73	99.31
Key-dep	25.51	32.26	42.61	57.43	75.31	90.61	98.11	99.86

Table 1: Success rate of both methods (given in %).

probability is:

$$P_{chosen.plaintext} = 2^9.\frac{1}{8}.(\frac{1}{16}.\frac{1}{8}.\frac{1}{8})^2.(\frac{12}{256}.\frac{1}{8}.\frac{3}{8}) = 2^{-22.83} \quad (11)$$

The next table indicates a possible complete set of equations and the keys suggested by every equation:

Equation	Keys suggested
$x_1 = 3y_3 + y_2 + 2y_1 + y_0$	0,2,13,15
$x_1 = y_3 + 2y_2 + 2y_1 + y_0$	1,3,5,7
$x_1 = y_3 + y_2 + 2y_1 + 3y_0$	4,6,9,11
$x_1 = 3y_3 + 2y_2 + 2y_1 + 3y_0$	8,10,12,14

For every key, these equations are correlated in the following way: the correct equation exhibits the best bias ($\pm 6/16$), one wrong equations exhibit a bias $\pm 2/16$ and the 2 last ones have a bias $\pm 0/16$.

From this, we can evaluate the success rate of both methods, as shown in Table 1, where N is the number of plaintext-ciphertext pairs. Our method is obviously faster because we only distinguish a 4-element set including the correct key, in place of the correct key itself in a classical linear cryptanalysis.

5.2 How to Find More Bits: The previous section underlines that we can quickly distinguish a 4-element set of keys using a key-dependent approximation of s-box SB. Then, the problem becomes to decrease the size of this key-set. Practically, this involves the use of other complete sets of equations and we illustrate this in Figure 5. For the input bit x_1 of SB, we found 12 equations, making up 3 complete sets (1,2,3) of equations. As every key is suggested by a different combination of these equations, we can recover the key as soon as all our 3 complete sets suggest the good 4-element set of keys. These 3 complete sets have the same success rate (say sr). Considering them as independent experiments, the probability that they all suggest the correct key would be sr^3. We observe that after a certain level of computations (about $N = 2^{42}$ in our example), this is more efficient that the classical key guess. Note that in practice, these complete sets are not independent[7] but we also found approximations involving other bits as we will explain in the next section. Another interesting point is the observation that some situations can never appear. Imagine that, after a certain level of computations, the first set suggests equation $1a$ and the second set suggests equation $2b$. As they have no common keys, we obviously know that one (or both) of these experiments are wrong.

[7]All these equations involve the same bits.

Key \ Equation	1a	1b	1c	1d	2a	2b	2c	2d	3a	3b	3c	3d
0	●				●				●			
1	●				●					●		
2		●				●					●	
3		●				●						●
4			●				●					●
5			●				●				●	
6				●				●		●		
7				●				●	●			
8				●	●				●			
9				●	●					●		
10			●			●					●	
11			●			●						●
12		●					●					●
13		●					●				●	
14	●							●		●		
15	●							●	●			

Figure 5: 3 Complete Sets of Equations involving x_1 and the Suggested Keys.

6. OPEN QUESTIONS AND IMPROVEMENTS

At this time, we have shown that the way to perform the final key guess in linear cryptanalysis does not necessarily involve the use of a real s-box. Situations where we use a non-linear key-dependent approximation to guess the key could be useful and at least, this method offers a great flexibility compared to the original one. The principal open problem is to investigate if this can help to make an efficient use of multiple approximations. This last point would be a great improvement as we demonstrate now.

In the preceding section, we investigated an efficient combination of 3 complete sets of equations, allowing to recover the 4 key-bits at the input of a non-linearly approximated s-box. These 12 equations all involve the same input and output bits. However, our exhaustive search allowed us to find other complete sets of equations, involving different input bits. As we suggest in section 4, it is also possible to find other linear approximations by simply changing the way we approximate SA and SB. Consequently, we can connect different linear approximations with our additional complete sets. Practically, for the same s-box SB, we found 9 complete sets of equations, including the 3 sets of Figure 5. As a result, we have a large redundancy in the information we get from a plaintext-ciphertext pair. The way we could combine this additional information is an open problem and of course, is very similar to the combination of multiple approximations in linear cryptanalysis, perhaps with a slightly different point of view.

Another open question concerns the choice of non-linear approximations. We investigated a particular case of quadratic relations, where our boolean functions can be represented as functions in $\mathbb{Z}_4$. Other approximations are possible and perhaps allow better results. Algebraic descriptions of block cipher components, as presented in [15], could for example be useful.

Finally, our investigations were limited to the final round of the block cipher Q. This allowed to evaluate a different way to guess the key. Previous papers presented different methods to use non-linear approximations in order to improve linear cryptanalysis. Most of these techniques could be combined with this work.

7. CONCLUSIONS

We explored a different way to perform the key guess in linear cryptanalysis. It allows great flexibility because the key is dynamically specified when additional plaintext-ciphertext pairs are provided. From a 4-element set including the correct key, we limit the set to 2 elements and finally recover the key itself.

In practice, we used a particular class of non-linear approximations to make up a system of equations whose solution is the secret key. An advantage is to be found in the large number of possible approximations (from 2^n to 4^n). As a consequence, we found a large number of equations that over-define the system. The best use of these multiple approximations is an open problem.

Compared to classical cryptanalysis, we improved the probability of success after a number of plaintext-ciphertext pairs is provided. We also have additional information because some solutions of the system are not allowed, informing the attacker about possible wrong experiments. Finally, we suggested a chosen-plaintext attack against the block cipher Q that reduced the required number of plaintext-ciphertext pairs from 2^{97} to 2^{87}.

REFERENCES

[1] J.Daemen and V.Rijmen, *AES Proposal: Rijndael*, http://csrc.nist.gov /encryption/aes/rijndael/Rijndael.pdf.

[2] R.Anderson, E.Biham, L.Knudsen, *Serpent : A Flexible Block Cipher With Maximum Assurance*, The First Advanced Encryption Standard Conference, Ventura, California, August 1998.

[3] L.McBride, *Q : A Proposal for NESSIE*, submitted to NESSIE, 2000.

[4] E.Biham et al, *Differential Cryptanalysis of Q*, in the proceedings of Fast Software Encryption 2001, LNCS 2355, pp.174-186, Springer-Verlag.

[5] L. Keliher, H. Meijer, and S. Tavares, *High Probability Linear Hulls in Q*, Proceedings of Second Open NESSIE Workshop, Royal Holloway College, University of London, Egham, U.K, 2001.

[6] L.Keliher, H.Meijer, S.Tavares, *New Method for Upper Bounding the Maximum Average Linear Hull Probability for SPNs*, proc. of Eurocrypt 2001, LNCS 2045, pp.420-436, Springer-Verlag.

[7] M.G.Parker, *Genralized s-box Nonlinearity*, NESSIE Report NES/DOC/UIB/ WP5/020/A, June 2002.

[8] H.Raadum, M.G.Parker, *Z4 Linear Cryptanalysis*, NESSIE Report NES/DOC/UIB/ WP5/018/1, June 2002.

[9] M.Matsui, *Linear Cryptanalysis Method for DES Cipher*, in the proceedings of Eurocrypt 93, LNCS 0765, pp.386-397, Springer-Verlag.

[10] K.Nyberg, *Linear Approximation of Block Ciphers*, in the proceedings of Eurocrypt 94, LNCS 0950, pp.439-444, Springer-Verlag.

[11] B.S.Kaliski, M.J.B.Robshaw, *Linear Cryptanalysis using Multiple Approximations*, in the proceedings of Crypto 94, LNCS 0839, pp.26-39, Springer-Verlag.

[12] L.R.Knudsen, M.J.B.Robshaw, *Non-Linear Approximations in Linear Cryptanalysis*, in the proceedings of Eurocrypt 96, LNCS 1070, pp.224-236, Springer-Verlag.

[13] T.Shimoyama, T.Kaneko, *Quadratic Relations of s-box and Its Application to the Linear Attack of Full Round DES*, in the proceedings of Crypto 98, LNCS 1462, pp.200-211, Springer-Verlag.

[14] L.R.Knudsen, J.E.Mathiassen, *A Chosen-Plaintext Linear Attack on DES*, in the proceedings of Fast Software Encryption 2000, LNCS 1978, pp.262-272, Springer-Verlag.

[15] A.Biryukov, C.De Canniere, B.Preneel, *Block Ciphers and Systems of Quadratic Equations*, in the proceedings of the Third NESSIE Workshop, November 6-7 2002, Munich, Germany.