Linear Cryptanalysis Using Multiple Approximations and FEAL

Burton S. Kaliski Jr. and M.J.B. Robshaw

RSA Laboratories
100 Marine Parkway
Redwood City, CA 94065, USA
burt@rsa.com matt@rsa.com

Abstract. We describe the results of experiments on the use of multiple approximations in a linear cryptanalytic attack on FEAL; we pay particular attention to FEAL-8. While these attacks on FEAL are interesting in their own right, many important and intriguing issues in the use of multiple approximations are brought to light.

1 Introduction

At Crypto'94 Matsui [5] presented details on experiments with linear cryptanalysis which derived the key used for the encryption of data with DES [11]. Such attacks on DES, however, still require too much known plaintext to be considered completely practical. The technique of simultaneously using multiple linear approximations, also presented at Crypto'94 [4], might be useful in reducing the amount of plaintext required for a successful linear cryptanalytic attack.

While initial experimental evidence has demonstrated the theoretical potential of using multiple approximations [4], it remains to be seen quite how powerful it might be in practice[1]. In this paper we shall describe the results of experiments on the use of multiple approximations in a linear cryptanalytic attack on the cipher FEAL [14], in particular on the eight-round version denoted FEAL-8.

In the following section we shall describe the essential features of the technique of linear cryptanalysis together with a description of how multiple linear approximations might be used. We shall then consider the linear cryptanalysis of FEAL and provide the results of various experiments we performed.

As well as describing a very recent improvement to these attacks, two issues relating to the use of multiple approximations are examined in the section that follows.

We are interested in seeing how the advantages gained by using multiple approximations compare to those obtained by the technique of *key ranking* which was introduced by Matsui [5]. Also, we describe the seemingly contradictory result that it is possible to improve the efficiency of an attack that uses three linear approximations by adding a fourth derived as the algebraic sum of the initial three. We close with our conclusions.

[1] Vaudenay has mentioned that multiple approximations can provide a factor of 64 reduction in the plaintext requirements for his work with variants of SAFER [15].

2 Linear cryptanalysis

2.1 Using a single approximation

Linear cryptanalysis is a technique which is proving to be very valuable in the analysis of block ciphers. While there are fascinating comparisons [2, 7, 10] to be made between linear cryptanalysis and the technique of differential cryptanalysis [3], linear cryptanalysis requires known rather than chosen plaintext and, as such, might well pose more of a practical threat to a block cipher than differential cryptanalysis.

Gradually the technique of linear cryptanalysis has been improved [6, 5] and in attacks on DES there is now a tantalizingly narrow gap between what can be achieved and what would be considered a practical cryptanalytic attack.

Linear cryptanalysis requires a linear approximation to the action of the block cipher. Such an approximation might be written as

$$P[\Gamma_1] \oplus C[\Gamma_2] = k_1$$

where $P[\Gamma_1]$ denotes the exclusive-or of specific bits of the plaintext P, $C[\Gamma_2]$ denotes the exclusive-or of certain bits of the ciphertext C and k_1 represents one bit of key information.

Following Matsui, it has become common practice to index bits of a plaintext or ciphertext block using zero to identify the rightmost bit; the bits of a 32-bit block are therefore numbered $31 \ldots 0$. When linear cryptanalysis is used on DES-like ciphers where the plaintext block is split into two halves, it is common to describe the left and right halves as high and low, P_H and P_L. We adopt similar notation for the ciphertext C.

A linear approximation will hold with some probability p. By taking a known plaintext/ciphertext pair, we obtain a guess for k_1 and provided $p \neq \frac{1}{2}$ we can use a simple algorithm (Matsui's *Algorithm 1* [6]) to decide the value of k_1. The more data we collect, the greater our confidence that we have correctly identified the value of k_1.

Of more practical importance are algorithms which solve for more than one bit of key information at a time. In particular, for iterated ciphers which repeatedly use the same round function, it is possible to use techniques which recover bits of the subkey used in either the first or last rounds of the cipher.

Typical attacks use a slightly different form of approximation to attack the subkey K^r used in the last round of an r-round cipher:

$$P_H[\Gamma_1] \oplus P_L[\Gamma_2] \oplus C_H[\Gamma_3] \oplus C_L[\Gamma_4] \oplus f(C_L, K^r)[\Gamma_3] = k_1.$$

To use this approximation we need to predict the value of $f(C_L, K^r)[\Gamma_3]$ with a high degree of certainty and Matsui [6, 5] has shown how this might be done. By analyzing the form of the round function it is possible to identify which bits of the subkey K^r and which bits of the input C_L effect the value of $f(C_L, K^r)[\Gamma_3]$. Suppose there are k such subkey bits, which are termed *effective key* bits and t relevant text bits, termed *effective text* bits.

We try each of the 2^k guesses for the effective key bits with all the data we have. When we have the correct guess for the effective key bits then the value of $f(C_L, K^r)[\Gamma_3]$ will be correct and linear cryptanalysis will continue as before. When we have an incorrect key guess we assume that the value of $f(C_L, K^r)[\Gamma_3]$ is '0' or '1' roughly equally often which results in the r-round approximation having a much reduced bias. Sufficient data is then taken to ensure that the correct guess can be distinguished from among the incorrect guesses thereby identifying the correct guess for the effective key bits.

Thus there is an algorithm (often referred to as Matsui's *Algorithm 2* [6]) which requires a basic computational effort of 2^{t+k} steps and can be used to recover the value of k bits of the subkey K^r and, in attacks on DES, the value of the single bit of key information k_1.

This technique of extending a round can theoretically be used on both the first and last rounds simultaneously. Two points about this approach are worth mentioning. First, when more guessing takes place more data is required to identify the correct guess. This, however, is usually more than offset by the fact that a better approximation is being used which itself requires less data for successful cryptanalysis. Second, the number of effective text and key bits increases making the required amount of computational effort for successful cryptanalysis more substantial.

2.2 Using multiple approximations

The authors have previously presented the idea of using several linear approximations simultaneously to reduce the amount of data required to mount a successful linear cryptanalytic attack [4].

We can most easily see how multiple approximations are used in an analog to *Algorithm 1*. Here we imagine that we have n linear approximations to the same bit of key information.

$$\begin{aligned} P_H[\Gamma_1^1] \oplus P_L[\Gamma_2^1] \oplus C_H[\Gamma_3^1] \oplus C_L[\Gamma_4^1] &= k_1 \\ P_H[\Gamma_1^2] \oplus P_L[\Gamma_2^2] \oplus C_H[\Gamma_3^2] \oplus C_L[\Gamma_4^2] &= k_1 \\ \vdots \\ P_H[\Gamma_1^n] \oplus P_L[\Gamma_2^n] \oplus C_H[\Gamma_3^n] \oplus C_L[\Gamma_4^n] &= k_1 \end{aligned}$$

Instead of getting one guess we get n guesses for the value of k_1 from each plaintext/ciphertext pair. A simple weighted sum for combining these guesses was proposed and it was shown that under certain natural assumptions, weights proportional to the absolute bias of the approximations provided the best results[2] [4]. Interestingly Murphy [9] has shown that under some alternative assumptions, different weights will provide the optimal results. In practice however, since the difference between these weighting schemes involves the bias as second-order

[2] We named the algorithm to do this *Algorithm 1M* thereby accentuating the fact that it is an extension of Matsui's *Algorithm 1* to allow the use of multiple approximations.

terms, the two weighting schemes are essentially the same when the approximations have very small bias.

There are now two practical issues which need to be overcome. First, we have to be sure that we can extend the use of multiple approximations to when we guess effective key bits in a round function. This is where the substantive work is done in a conventional linear cryptanalytic attack and without a similar technique for multiple approximations, the effectiveness of multiple approximations will be limited.

Second, it seems to be very unlikely that for a good block cipher, several linear approximations each with a good bias can be identified, all providing approximations to the same bit of key information. Ideally we need a technique which allows us to use several approximations simultaneously even if they involve approximations to different bits of key information.

For the first problem it is easy to draw up an analogous algorithm to Matsui's *Algorithm 2*. If the linear approximations require different effective key and text bits then when we use all n linear approximations simultaneously the set of effective key and text bits becomes equal to the union of all sets of effective key and text bits for each approximation. Thus, to ensure that there is no increase in the work effort due to an increased number of effective bits, each approximation must use the same effective key and text bits.

For the second problem we need only guess the relation between the bits of key information involved [4]. For instance, to use two linear approximations

$$P_H[\Gamma_1^1] \oplus P_L[\Gamma_2^1] \oplus C_H[\Gamma_3^1] \oplus C_L[\Gamma_4^1] = k_1$$
$$P_H[\Gamma_1^2] \oplus P_L[\Gamma_2^2] \oplus C_H[\Gamma_3^2] \oplus C_L[\Gamma_4^2] = k_2,$$

we would guess whether $k_1 = k_2$ or $k_1 \neq k_2$. We would then pursue the rest of our analysis under each assumption in turn (thereby potentially doubling the work effort for any later stages of analysis) until we are in a position to reject one of the options.

We note, however, that recent work has established a more efficient approach when multiple approximations are used to recover the effective key bits in some additional round. We shall discuss this new approach in Section 4.2 where we replace the original *Algorithm 2M* which required the use of approximations to the same bit of key information [4], with a much more general and useful algorithm.

3 Linear cryptanalysis and FEAL

Techniques that lie at the heart of linear cryptanalysis were originally used by Matsui and Yamagishi to attack small round versions of FEAL [8]. Central to these attacks is a rewriting of FEAL to give an equivalent cipher; we shall use this technique in the attack presented here.

There has been considerable recent work completed on the linear cryptanalysis of FEAL-8 [2, 1, 12, 13]. Matsui and Yamagishi [8] originally showed that

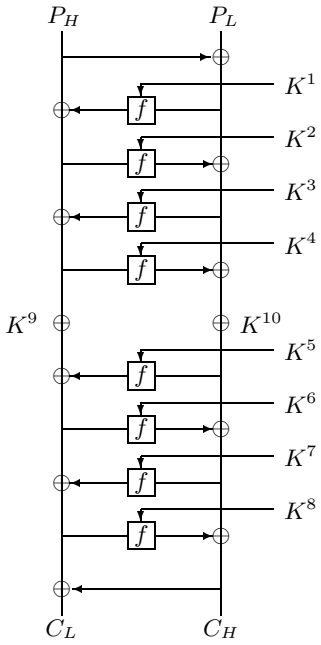

Fig. 1. Modified FEAL-8

there were attacks requiring 2^{28} or 2^{15} known plaintexts but with an infeasible computational workload. Biham [2] discovered a linear approximation which could be used to attack FEAL-8 with 2^{24} known plaintexts and an estimated success rate of 78% and Aoki et al. [1] describe an attack requiring 2^{25} known plaintexts for a success rate of more than 70%. We decided to closely follow the work of Aoki et al. who provide a very detailed account of their attack. In this paper we show how multiple approximations can be used very easily in two vital stages to improve the attack on FEAL-8 by almost a factor of four.

3.1 Modified FEAL

Matsui and Yamagishi [8] have shown how FEAL can be rewritten so that it is more amenable to standard linear cryptanalytic techniques.

In the original specification of FEAL [14], key material is exclusive-ored with the data that enters a Feistel network and with that leaving. However, this key material can be moved into the Feistel network provided we change the definition of the round functions to allow the introduction of additional key material. By moving key material from both ends of the Feistel network into the middle, and by assuming that all the key material is independent, FEAL-8 can be written as shown in Figure 1. The round functions for this modified version of FEAL are

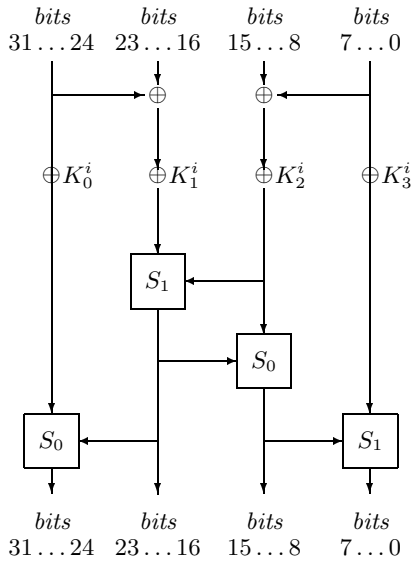

Fig. 2. The round function $f(\cdot, K^i)$ in modified FEAL. The S-boxes are defined by $S_0(x, y) = x + y \bmod 256$ and $S_1(x, y) = x + y + 1 \bmod 256$.

presented in Figure 2.

3.2 Practical issues

The attack of Aoki et al. [1] is very involved and highlights many of the complications to be found in mounting a linear cryptanalytic attack on FEAL. There are seven stages to this attack in which the first six are used to identify bits of the subkey in the first and final rounds until one round can be removed leaving the cryptanalyst with the task of attacking FEAL-7. There is an attack on FEAL-7 [8] requiring 2^{15} known plaintexts and so we find that the data requirements for the attack on FEAL-8 are dominated by those for the first six stages.

The reader is referred to the work of Aoki et al. for more detail, but in summary the requirements for the various rounds are provided in Table 1 of this paper. From the table presented there, it is clear that the most data intensive stages are the first and the third. We will see in the next section that we are able to improve the attack on FEAL-8 using multiple approximations in only two stages; fortuitously these are the first and third stages. As a consequence, we are able to lower the overall data requirements for a linear cryptanalytic attack on FEAL-8.

Before moving on to the details of using multiple approximations we mention some of the difficulties of using conventional linear cryptanalysis on FEAL; many

of these issues have already been covered elsewhere [1, 13].

As we have previously stressed, the success of linear cryptanalysis almost certainly depends on being able to recover more than one bit of key information with a single approximation. And, as previously described, the usual technique is to recover what are termed effective key bits from the subkey used in either the first or last round or, if possible, both rounds simultaneously.

In DES, the parallel structure of the S-boxes ensures that the number of effective key and text bits can be minimized (to six) by considering approximations that use a single S-box. As we can see from Figure 2, in considering the output of a single S-box in FEAL, we are forced to consider at least 16 bits of subkey, usually more.

In practice these features serve to increase the number of effective text and key bits in any one round. In fact, for the approximations we currently have available, the number of these effective bits is so large that it is computationally infeasible to search over the effective key bits required for an attack which attempts to remove *both* the first and final rounds simultaneously. Thus, this very useful technique in attacking DES is practically infeasible in attacking FEAL where we can only remove one round. In addition, attacks on DES recover the value of the internal bit of key information in addition to the effective key bits in the outer rounds; with FEAL we only recover the effective key bits.

3.3 Using multiple approximations

As an example of our approach we shall closely consider the first phase of the attack due to Aoki et al. We must note that because of slight differences between FEAL and usual Feistel networks, the input to the round function used in the last round is effectively $C_H \oplus C_L$ rather than C_L.

In phase one, the following single approximation A_1, an extension of one discovered by Biham [2], is used:

$$\begin{aligned} P_L[16,23,25,26,31] \oplus C_H[31] \oplus C_L[16,23,25,26] \\ \oplus f(C_H \oplus C_L, K^8)[23,25,31] = k_1. \end{aligned}$$

The probability that this approximation holds for randomly chosen plaintext is $p = \frac{1}{2} + 2^{-11}$ and so, following Matsui [6], it might be expected that by taking $8 \times \epsilon^{-2} = 2^{25}$ known plaintexts, where the *bias* ϵ satisfies $p = \frac{1}{2} + \epsilon$, the effective key bits of K^8 can be derived with a good success rate.

In a similar experiment on eight-round DES this good success rate was 99% [6], but with FEAL-8 there is a slight degradation in the quality of the results; Aoki et al. have experimentally established that the success rate is about 78% while our own experiments (with 50 trials) reveal a success rate of 88%. This slight degradation might be due to the increased number of effective key bits we must guess in an attack on FEAL or perhaps there is some intrinsic feature of FEAL that distinguishes it from DES. Either way, we should be cautious in using algorithm performance estimates obtained for DES in estimating the effectiveness of our attacks on FEAL.

It is not difficult to identify another three linear approximations A_2, A_3, A_4 with the same bias as A_1:

$$\begin{aligned}
P_H[24,31] \oplus P_L[16,23,25,26,31] &\oplus C_H[24] \oplus C_L[16,23,24,25,26,31] \\
&\oplus f(C_H \oplus C_L, K^8)[23,25,31] = k_2 \\
P_H[22,24] \oplus P_L[16,23,25,26,31] &\oplus C_H[22,24,31] \oplus C_L[16,22,23,24,25,26] \\
&\oplus f(C_H \oplus C_L, K^8)[23,25,31] = k_3 \\
P_H[22,31] \oplus P_L[16,23,25,26,31] &\oplus C_H[22] \oplus C_L[16,22,23,25,26,31] \\
&\oplus f(C_H \oplus C_L, K^8)[23,25,31] = k_1 \oplus k_2 \oplus k_3
\end{aligned}$$

There are two issues to consider when we use all four approximations. First, the bits of key information k_1, k_2 and k_3 differ only in bits of the subkey K^9. By using the technique of guessing the relation between these bits, or by using our new algorithm (Section 4.2), we can use all four simultaneously.

Second, we note that there is no increase in the number of effective key bits when we use all four of these approximations instead of one. This is because all four cases share the term $f(C_H \oplus C_L, K^8)[23,25,31]$ and so the same effective key and text bits are used for all four approximations.

As Aoki et al. previously discovered [1] there are many technical issues to resolve in identifying which bits of the subkey K^8 are useful to include as effective key bits and which bits of the subkey K^8 can actually be recovered with any degree of certainty.

Some bits of the subkey K^8 have little impact on the value of $f(C_H \oplus C_L, K^8)[23,25,31]$ and to minimize the computational effort, we ignore these bits totally. Another complication is that some bits of the subkey merely complement the value of $f(C_H \oplus C_L, K^8)[23,25,31]$ when they themselves are complemented; one example of this is bit 31 in the subkey K^8. With the algorithms at our disposal, we would be unable to distinguish between a guess for the effective bits of K^8 when bit 31 is set to '0' and when it is set to '1'.

In addition, there is a problem in differentiating between a set of effective key bits and a related set where every bit in the original set is complemented. To combat this we follow the example of Aoki et al. [1] and recover the value of each effective key bit exclusive-ored with bit 30 (which was chosen arbitrarily) of the subkey K^8. Note that Aoki et al. also describe the effective key bits as *explored* key bits, which are needed for successful analysis, and *detected* key bits, which are actually recovered; we do not, however, use this terminology here. Note that the effective text referred to here is the data entering the S-boxes after the initial exclusive-or in the round function.

In the following table we list the effective text and key bits we used, together with the bits[3] of subkey K^8 we were able to recover.

effective text bits	$8,9,10,11,16,17,18,19,26,27,28,29,30,31$ *and* $12 \oplus 20$
effective key bits	$\{8,9,10,11,16,17,18,19,26,27,28,29\} \oplus 30$ *and* $12 \oplus 20$
recovered key bits	$\{9,10,11,17,18,19,28,29\} \oplus 30$ *and* $12 \oplus 20$

[3] We recover $12 \oplus 20$ as a bit of key information rather than $12 \oplus 20 \oplus 30$ which is recovered by Aoki et al. [1].

		Aoki et al.		*multiple approximations*	
phase	*key attacked*	*plaintexts*	*success rate*	*plaintexts*	*success rate*
1	K^8	2^{25}	79%	2^{23}	81%
2	K^1	2^{24}	100%	2^{23}	91%
3	K^1	2^{25}	91%	2^{23}	91%
4	K^1	2^{20}	100%	2^{20}	100%
5	K^8	2^{24}	100%	2^{23}	100%
6	K^8	2^{17}	100%	2^{17}	100%
7	*FEAL-7*	2^{15}	100%	2^{15}	100%
full FEAL-8		2^{25}	72%	2^{23}	67%

Table 1. The success rates for different stages of the attack of Aoki et al. on FEAL-8. Changes to phases one and three are the direct result of using four linear approximations; the success rate for phase one is derived experimentally and the success rate for phase three is the theoretical prediction. Success rates for phases two and five are those provided by Aoki et al. for a reduced number of plaintexts.

While we have concentrated our attention on the first phase of the attack due to Aoki et al. we note that very similar techniques can be used to devise a similar modification to the third phase. Aoki et al. use one approximation

$$P_H[7] \oplus P_L[1,2,8,15] \oplus C_L[1,2,7,8,15] \oplus \\ \oplus f(P_H \oplus P_L, K^1)[1,7,15] = k_1.$$

We would use the following three linear approximations as well:

$$\begin{aligned} P_H[0] \oplus P_L[0,1,2,7,8,15] \oplus C_L[1,2,7,8,15] \oplus C_H[0,7] \\ \oplus f(P_H \oplus P_L, K^1)[1,7,15] = k_2, \\ P_H[0,7,14] \oplus P_L[0,1,2,8,14,15] \oplus C_L[1,2,7,8,15] \oplus C_H[0,14] \\ \oplus f(P_H \oplus P_L, K^1)[1,7,15] = k_3, \\ P_H[14] \oplus P_L[1,2,7,8,14,15] \oplus C_L[1,2,7,8,15] \oplus C_H[7,14] \\ \oplus f(P_H \oplus P_L, K^1)[1,7,15] = k_1 \oplus k_2 \oplus k_3. \end{aligned}$$

3.4 Experimental results

Most of the experiments we conducted were performed on FEAL-4 with some confirmatory experiments completed using FEAL-8. All the techniques we have described for FEAL-8 can easily be converted to attacks on FEAL-4 with little or no variation except for the bias of the approximations (without regard for sign) which increases from 2^{-11} to 2^{-5}, and of course, the number of subkeys involved.

As we increased the number of linear approximations in the first phase of the analogous attack on FEAL-4 we obtained the following results. All success rates are quoted for 100 trials.

	number of plaintexts					
	$1,024$	$2,048$	$4,096$	$8,192$	$16,384$	$32,768$
A_1	3%	16%	45%	79%	100%	100%
A_1, A_2	13%	35%	82%	100%	–	–
A_1, A_2, A_3	23%	60%	96%	100%	–	–
A_1, A_2, A_3, A_4	46%	73%	99%	100%	–	–

It is easy to see that using four linear approximations allows for a factor of four reduction in the plaintext required for a successful linear cryptanalytic attack. When we turn to FEAL-8 we find that when using one linear approximation our experiments give a better success rate than that reported by Aoki et al. With 2^{25} known plaintexts we found that we could attack FEAL-8 using one linear approximation with a success rate of 88%. With 2^{26} known plaintexts this success rate increased to 98%. By using four linear approximations together, we have been able to attack FEAL-8 in experiments using 2^{23} known plaintexts with a success rate of 81% and using 2^{24} known plaintexts with a success rate of 99%.

One point we make here is that if we guess the relation between these four approximations, we force ourselves to consider various alternative results to this first phase of the attack. We would therefore continue with each of these alternatives until we can discount them. In our variant of the attack due to Aoki et al. this would mean an increased work factor of up to four times for stages two and three and of up to 16 times for stages following the third. However, results in Section 4.2 show that we can avoid this increase in the work effort *and* recover more bits of key information.

As a result of our experiments it is reasonable to conclude that the attack of Aoki et al. can be improved in terms of the amount of plaintext required using multiple approximations. Where Aoki et al. require 2^{25} known plaintexts for a 72% success rate, the use of multiple approximations should provide an attack requiring 2^{24} known plaintexts for a success rate that is very close to 99%.

Using other figures provided by Aoki et al. for the success rate of the second and fifth phases of their attack when 2^{23} known plaintexts are used, and our own experimental results for the first phase, we anticipate that FEAL-8 is vulnerable to linear cryptanalysis using 2^{23} known plaintexts with a success rate of 67%. These results are summarized in Table 1.

4 New developments and some open issues

In this section we consider some issues relevant to the general use of multiple approximations. While our observations in this section are likely to be more generally significant than those of the previous section, it is only by implementing our attack against FEAL that these issues have come to the fore. Much of our work in this section is preliminary and is still the subject of ongoing research.

4.1 Key ranking

At Crypto'94 Matsui presented the first experimental cryptanalysis of DES [5]. The innovative feature of this attack which allows an important reduction in plaintext requirements is the idea of what we shall term *key ranking*.

When attempting to identify the correct guess for a set of effective key bits, original techniques use a scoring system; the guess with the highest score is considered to be the most likely to be correct. If instead, the cryptanalyst takes the ten guesses with the highest scores (for instance) then by continuing analysis with all ten guesses the success rate will increase, or correspondingly, the same success rate will be achieved with a smaller number of known plaintexts.

This then raises a question. We have two independent techniques for reducing the amount of plaintext required for a successful attack on some cipher. Will the advantages gained by using key ranking be greater than those gained by using multiple approximations? Further, can both techniques be used together to provide an even greater advantage?

We performed some tests on FEAL-4 to see how these questions might be answered in this particular case. To compare key ranking and the use of four multiple approximations we took the four guesses with the highest score resulting from our analysis and noted how often the correct key value fell within these four[4]. The success rates for these experiments are provided below.

	number of plaintexts			
	$1,024$	$2,048$	$4,096$	$8,192$
A_1 *only*	3%	16%	45%	79%
A_1 *with ranking*	11%	31%	70%	97%
A_1, A_2, A_3, A_4	46%	73%	99%	100%
$A_1 \ldots A_4$ *with ranking*	69%	96%	100%	100%

We note that using both techniques together, multiple approximations and key ranking, we still get an improvement in performance over and above that for either method in isolation. In fact this behavior might be viewed as a natural consequence of work by Murphy et al. [10] and it suggests that any linear cryptanalytic attacks which use key ranking for enhanced performance might still be improved further when multiple approximations are used.

4.2 *Algorithm 2MG*

In this section we report on a recent improvement to the technique of using multiple linear approximations to identify the effective key bits of some subkey in the outer rounds of a cipher. When we first introduced the use of several linear approximations, we stated that with approximations to different bits of

[4] Obviously we could have chosen more than four guesses with the highest score, but for comparison with multiple approximation techniques we chose four so that the same work effort would be required as when guessing the relation between the key bits in the four linear approximations we used.

key information we would have to guess the relation between the different bits in order to use all the approximations simultaneously. This would then lead to an increase in the work effort for subsequent phases of analysis.

But we have found that when we are using multiple approximations to derive the value of some set of effective key bits, it is possible to obtain the value of the bits of key information in the different approximations at the same time as deriving the value of the effective key bits.

We shall present an algorithm, *Algorithm 2MG*, to accomplish this. It is, in fact, a more general version of an algorithm presented as part of earlier work [4]. To give the algorithm in its most general form, we shall assume that we are attempting to identify the value of some effective key bits in both the first and last rounds of some cipher. We shall also assume that we are attacking a basic Feistel network. It is trivial to modify the form of the algorithm to suit our attack on FEAL. We note that there are several optimizations to the basic outline of *Algorithm 2MG* which might well be beneficial for implementation.

For an r-round Feistel cipher we approximate $(r-2)$ iterations of the round function f from the second to the $(r-1)^{\text{th}}$ round using n linear approximations while we still make guesses for the subkey bits needed to extend through the first and final rounds. Note that for practical reasons, the approximations involve the same guessed subkey bits in round one, as well as in round r. Following earlier notation we can write the i^{th} linear approximation as follows:

$$\begin{aligned} P_H[\Gamma_1^i] \oplus P_L[\Gamma_2^i] \oplus C_H[\Gamma_3^i] \oplus C_L[\Gamma_4^i] \\ \oplus f(P_L, K^1)[\Gamma_1^i] \oplus f(C_L, K^r)[\Gamma_3^i] = k_i. \end{aligned} \tag{1}$$

We use k_i for $1 \leq i \leq n$ to denote the bit of key information in linear approximation i. We will suppose, without loss of generality, that the probability p_i that each approximation holds is greater than $\frac{1}{2}$. Recall that we define the *bias* ϵ_i of each approximation as $\epsilon_i = |p_i - \frac{1}{2}|$.

Step 1 Let $K^1[g]$ $(g = 1, 2, \ldots)$ and $K^r[h]$ $(h = 1, 2, \ldots)$ be possible candidates for the effective bits of subkeys K^1 and K^r respectively. Then for each pair $(K^1[g], K^r[h])$ and each linear approximation i, let $T_{g,h}^i$ be the number of plaintexts such that the left side of equation 1 is equal to 0 when K^1 is replaced by $K^1[g]$ and K^r by $K^r[h]$. Let N be the total number of plaintexts.

Step 2 Let $a_i = \epsilon_i / \sum_{i=1}^n \epsilon_i$. Define the n-tuple[5] $C = (c_1, \ldots c_n)$. Calculate for each g, h and each C,

$$U_{g,h}[C] = \sum_{\substack{i=1 \\ c_i=0}}^{n} a_i T_{g,h}^i + \sum_{\substack{i=1 \\ c_i=1}}^{n} a_i (N - T_{g,h}^i)$$

Step 3 Let U_{max} be the maximum value of all $U_{g,h}[C]$'s.

- Adopt the key candidate corresponding to U_{max} and guess $k_i = c_i$ for $1 \leq i \leq n$.

[5] This tuple represents the possible values for the bits of key information in each approximation.

Intriguingly, very little additional plaintext is required to recover these additional bits of key information. We performed experiments using two linear approximations in an attack on FEAL-4 and in an analogy to the previous method of guessing the relation between the approximated bits of key information, we derive the relation between these bits of key information.

In our experiments we get a very similar success rate with the same number of known plaintexts when we derive rather than guess the relation between the relevant bits of key information. These experiments were performed on different sets of data, explaining the slightly increased success rate for $2,048$ known plaintexts.

	number of plaintexts			
	$1,024$	$2,048$	$4,096$	$8,192$
A_1, A_2 *and guessing the relation*	13%	35%	82%	100%
A_1, A_2 *and deriving the relation*	13%	36%	75%	99%

The price we pay in using this new *Algorithm 2MG* is an increase in the amount of calculation at the time of analysis. However, there is no additional work effort for subsequent rounds. In effect, when using n linear approximations to different bits of key information, we perform linear cryptanalysis n times, but combine the results up to 2^{n-1} times, using each possible relation between the key information in the n approximations.

We then take, using a simple scoring system, the guess with the highest score from among all these possible guesses. By taking sufficient plaintext it is possible to recover both the correct guess for the effective key bits *and* the correct value of the key bits used in the approximations.

Using this enhancement in our attack on FEAL-8 we would expect the first and third phases to take four times as long as they would when using a single approximation, but there would be no increase in the work effort for subsequent phases. This gives one major advantage in the use of multiple approximations over key ranking, though we must note that this is only for this particular attack on FEAL-8.

Since we are identifying more bits of key information, which therefore require more guesses, we would expect a slight diminution in the success rate of our attack when using the same amount of known plaintext. However, as we can see in this simple case of two approximations in an attack on FEAL-4, this diminution might be very slight.

4.3 Linear independence of approximations

One very interesting development in the use of multiple approximations here is, at first sight, counter-intuitive. In our attack on FEAL-8 we used four linear approximations $A_1, \ldots, A_4$. It can easily be verified however, that A_4 is the algebraic sum of the other three approximations. In other words, the four linear approximations are not algebraically linear independent.

So why did we get an improved performance when we considered four linear approximations instead of three? Where is the additional information coming from?

We don't believe that it is a question of extra information, rather, the use of the fourth approximation with our techniques extracts more of the information that is already available when using three approximations.

When considering three linear approximations, information can be gained from each in turn, but there is other information which is revealed if we consider the approximations jointly. It is this additional information which our original techniques fail to extract.

At present, with three linear approximations, we use a rather simple scoring system to obtain the best candidate for the effective key bits. The action of adding the fourth approximation in our attack on FEAL-4, could equally have been simulated by modifying the simple scoring system used with the three approximations. And, as we saw in Section 3.4, this modified scoring system can extract more information than the previous simple one.

There is now considerable work to be done in deciding quite how we can extract the most information from the data we collect in an attack. While it may be the case that such 'optimal' techniques remain specific to the block cipher under attack, it is hoped that some general results will show the way to less data-intensive linear cryptanalytic attacks.

5 Conclusions

In this paper we have described an improvement to current linear cryptanalytic attacks on FEAL-8 by using multiple approximations. We have also confirmed experimentally some of the estimates made using current techniques and we have outlined an attack on FEAL-8 which requires 2^{23} known plaintexts for an expected success rate of 67% or 2^{24} known plaintexts for an expected success rate of 99%.

We also used FEAL-4 as a test-bed for some more interesting and, perhaps ultimately, more important experiments.

First, we showed that it is not difficult to use multiple approximations which differ in the key information they approximate. Not only that, but we have presented a new algorithm, *Algorithm 2MG*, which allows more key information to be recovered when multiple approximations are used, in addition to reducing the plaintext requirements for a successful attack.

Second, we have shown that the gains made using multiple approximations might be comparable, if not better, than those obtained using the technique of key ranking. In fact, we believe that both techniques can be used together to provide improvements over and above those available using either technique alone.

Finally, we have highlighted the somewhat counter-intuitive result that even if the linear approximations we use are not algebraically linearly independent, they might still all be used simultaneously to beneficial effect.

The full implications of all these developments are the subject of continuing research. But, under circumstances which are closely dependent on the block cipher, the use of multiple approximations in a linear cryptanalytic attack can be expected to be a valuable technique. More generally, we believe this work has demonstrated that the full potential of this relatively new type of cryptanalysis is yet to be realized.

Acknowledgement We would like to thank Sean Murphy for some interesting and helpful comments.

References

1. K. Aoki, K. Ohta, S. Araki, and M. Matsui. *Linear Cryptanalysis of FEAL-8 (Experimentation Report).* Technical Report ISEC 94-6 (1994-05), IEICE, 1994.
2. E. Biham. On Matsui's linear cryptanalysis. In *Advances in Cryptology — Eurocrypt '94*, Springer-Verlag, to appear.
3. E. Biham and A. Shamir. *Differential Cryptanalysis of the Data Encryption Standard.* Springer-Verlag, New York, 1993.
4. B.S. Kaliski Jr. and M.J.B. Robshaw. Linear cryptanalysis using multiple approximations. In Y.G. Desmedt, editor, *Advances in Cryptology — Crypto '94*, pages 26–39, Springer Verlag, New York, 1994.
5. M. Matsui. The first experimental cryptanalysis of the Data Encryption Standard. In Y. G. Desmedt, editor, *Advances in Cryptology — Crypto '94*, pages 1–11, Springer-Verlag, New York, 1994.
6. M. Matsui. Linear cryptanalysis method for DES cipher. In T. Helleseth, editor, *Advances in Cryptology — Eurocrypt '93*, pages 386–397, Springer-Verlag, Berlin, 1994.
7. M. Matsui. On correlation between the order of the S-boxes and the strength of DES. In *Advances in Cryptology — Eurocrypt '94*, Springer-Verlag, to appear.
8. M. Matsui and A. Yamagishi. A new method for known plaintext attack of FEAL cipher. In R.A. Rueppel, editor, *Advances in Cryptology — Eurocrypt '92*, pages 81–91, Springer-Verlag, Berlin, 1992.
9. S. Murphy. August 1994. Personal communication.
10. S. Murphy, F. Piper, M. Walker and P. Wild. Likelihood estimation for block cipher keys. May 1994. Preprint.
11. National Institute of Standards and Technology (NIST). *FIPS Publication 46-2: Data Encryption Standard.* December 30, 1993.
12. K. Ohta and K. Aoki. *Linear Cryptanalysis of the Fast Data Encipherment Algorithm.* Technical Report ISEC 94-5 (1994-05), IEICE, 1994.
13. K. Ohta and K. Aoki. Linear cryptanalysis of the fast data encipherment algorithm. In Y. G. Desmedt, editor, *Advances in Cryptology — Crypto '94*, pages 12–16, Springer-Verlag, New York, 1994.
14. A. Shimizu and S. Miyaguchi. Fast data encipherment algorithm FEAL. In D. Chaum and W.L. Price, editors, *Advances in Cryptology — Eurocrypt '87*, pages 267–280, Springer-Verlag, Berlin, 1988.
15. S. Vaudenay. On the need for multipermutations: Cryptanalysis of MD4 and SAFER. In these proceedings.

This article was processed using the LaTeX macro package with LLNCS style