On the Security of the IDEA Block Cipher *

Willi Meier

HTL Brugg-Windisch, CH-5200 Windisch, Switzerland
email: meierw@htlulx.htl-bw.ch

Abstract. IDEA is an iterated block cipher proposed by Lai and Massey and is based on the design concept of "mixing operations from different algebraic groups". New arithmetic properties of the basic operations used in the round function are found and investigated with respect to the security of this block cipher. Evidence is given that these properties can be exploited in the cryptanalysis of the first 2 rounds of IDEA but that they are of no assistance in the cryptanalysis of the full IDEA block cipher containing 8 rounds.

1 Introduction

In [3] J. Massey and X. Lai introduced a new iterated block cipher, the Proposed Encryption Standard (PES). The differential cryptanalysis of PES carried out in [4] suggested a minor modification, called Improved PES (IPES). It was shown in [4] that this modification of PES improves the security against differential cryptanalysis. In recent work of Lai [5], the modified block cipher IPES is named IDEA (International Data Encryption Algorithm). The IDEA contains 8 computationally identical rounds plus an output transformation. The plaintext and the ciphertext are 64 bit blocks, while the secret key is 128 bit long. The cipher is based on the design concept of "mixing (arithmetic) operations from different algebraic groups".

Our aim is to contribute to a systematic investigation of arithmetic properties of both the basic operations and the round function of IDEA with respect to the security of this block cipher. The basic operations used in the design are multiplication modulo $2^{16}+1$ (where 0 is taken as 2^{16}), integer addition modulo 2^{16}, and bit-by-bit exclusive-OR of two 16 bit subblocks. In [3] the interaction of these operations is studied as it contributes to the "confusion" required for a secure cipher. In particular, it is stated in [3], that the 3 operations are incompatible in the sense that no pair out of them satisfies a distributive law. In Section 3 we shall show however, that the multiplication and the integer addition satisfy a "partial" distributive law, stemming from arithmetic modulo $2^{16}+1$. This fact made a detailed investigation necessary and may also be of interest for the construction of other cryptographic algorithms based on arithmetic operations. The interaction of the group operations is further studied in Section 4, where arithmetic properties in the context of a class of one-round differentials are in-

* This work is supported by Stiftung Hasler-Werke, Switzerland

T. Helleseth (Ed.): Advances in Cryptology - EUROCRYPT '93, LNCS 765, pp. 371-385, 1994.

vestigated. Our considerations extend a result in [5] and are useful in the cryptanalysis of few rounds of IDEA in Section 5. We give estimates for the computational complexity to break the first few rounds by combining results related to one-round differentials and the partial distributive law. We give evidence that the newly found arithmetic properties can be exploited in the cryptanalysis of the first 2 rounds of IDEA, but that they are of no assistance in the cryptyanalysis of a block cipher containing 3 or more rounds of IDEA. This estimate fits nicely with a conclusion drawn in [5] saying that IDEA will be secure against a differential cryptanalysis attack after only 4 of its 8 rounds.

2 Description of IDEA

For our analysis we recall the description of the IDEA-algorithm as given in [4] and [5]. In the block cipher IDEA (International Data Encryption Algorithm) plaintext and ciphertext are 64 bit blocks and the key is 128 bits long. The cipher is based on a novel design concept of mixing different arithmetic operations rather than using boolean functions (e.g., in terms of lookup tables). The cipher structure is chosen to provide confusion and diffusion and to facilitate both hardware and software implementation. For the latter aspect we refer to [2].

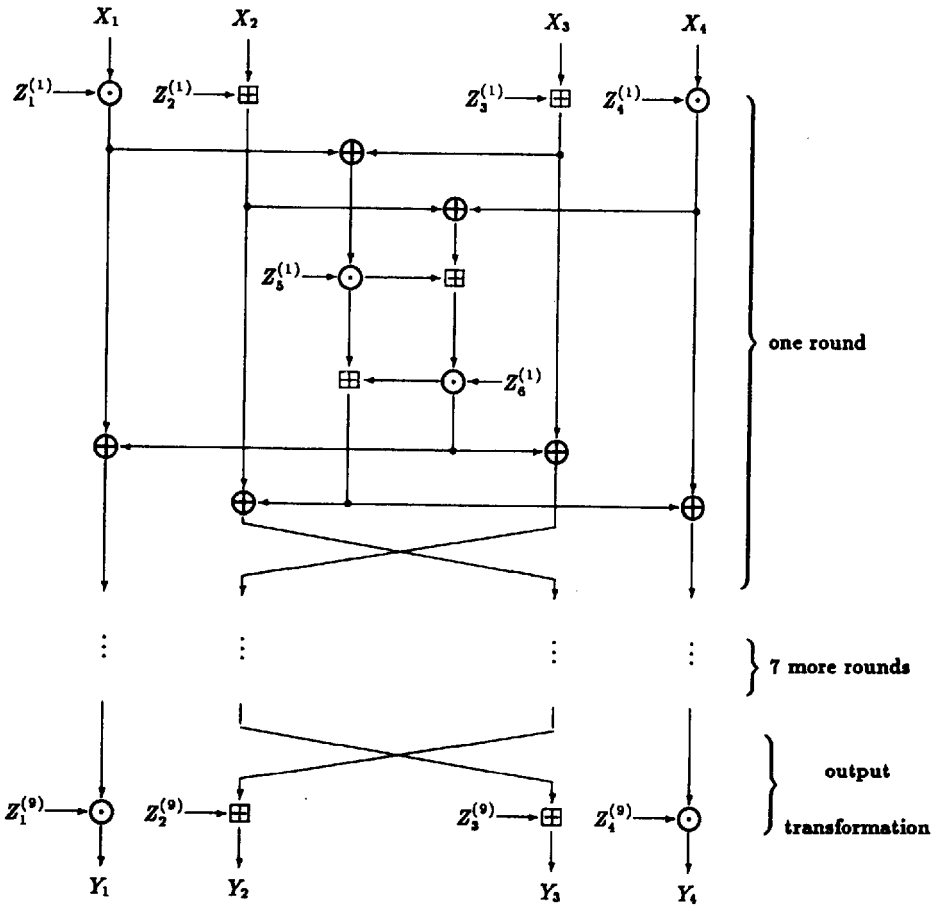
The IDEA-algorithm is an iterated cipher consisting of 8 computationally identical rounds followed by an output transformation. The complete first round as well as the output transformation are depicted explicitly in the computational graph shown in Figure 1.

2.1 Encryption

In the encryption process, three different (arithmetic) group operations on pairs of 16-bit subblocks are used, namely

- bit-by-bit exclusive-OR of two 16 bit subblocks, denoted as $\oplus$;
- addition of integers modulo 2^{16} where the 16 bit subblock is treated as the usual radix-two representation of an integer; the resulting operation is denoted as $\boxplus$;
- multiplication of integers modulo $2^{16}+1$ where the 16 bit subblock is treated as the usual radix-two representation of an integer except that the all-zero subblock is treated as representing 2^{16}; the resulting operation is denoted as $\odot$.

The 64 bit plaintext block X is partitioned into four 16 bit subblocks X_1, X_2, X_3, X_4, i.e., $X = (X_1, X_2, X_3, X_4)$. The four plaintext subblocks are transformed into the four 16 bit ciphertext subblocks Y_1, Y_2, Y_3, Y_4 under the control of 52 key subblocks of 16 bits that are formed from the 128 bit secret key to be described in the key schedule. For $r = 1, 2, ..., 8$, the six key subblocks used in the r-th round are denoted as $Z_1^{(r)}, Z_2^{(r)}, ..., Z_6^{(r)}$. Four 16 bit key subblocks are used in the output transformation; these subblocks are denoted as $Z_1^{(9)}, Z_2^{(9)}, Z_3^{(9)}, Z_4^{(9)}$.



X_i : 16-bit plaintext subblock
Y_i : 16-bit ciphertext subblock
$Z_i^{(r)}$: 16-bit key subblock
$\oplus$: bit-by-bit exclusive-OR of 16-bit subblocks
$\boxplus$: addition modulo 2^{16} of 16-bit integers
$\odot$: multiplication modulo $2^{16}+1$ of 16-bit integers with the zero subblock corresponding to 2^{16}

Fig. 1. Computational graph for the encryption process of the IDEA cipher.

2.2 The decryption process

The computational graph of the decryption process is essentially the same as that for encryption, the only change being that the decryption key subblocks $K_i^{(r)}$ are computed from the encryption key subblocks $Z_i^{(r)}$ as follows:

$$(K_1^{(r)}, K_2^{(r)}, K_3^{(r)}, K_4^{(r)}) = (Z_1^{(10-r)^{-1}}, -Z_3^{(10-r)}, -Z_2^{(10-r)}, Z_4^{(10-r)^{-1}})$$
$$\text{for r = 2,3,...,8;}$$
$$(K_1^{(r)}, K_2^{(r)}, K_3^{(r)}, K_4^{(r)}) = (Z_1^{(10-r)^{-1}}, -Z_2^{(10-r)}, -Z_3^{(10-r)}, Z_4^{(10-r)^{-1}})$$
$$\text{for r = 1 and 9;}$$
$$(K_5^{(r)}, K_6^{(r)}) = (Z_5^{(r)}, Z_6^{(r)}) \quad \text{for r=1,2,...,8;}$$

where Z^{-1} denotes the multiplicative inverse (modulo $2^{16}+1$) of Z, i. e., $Z \odot Z^{-1} = 1$ and $-Z$ denotes the additive inverse (modulo 2^{16}) of Z, i. e., $-Z \boxplus Z = 0$.

2.3 The key schedule

The 52 key subblocks of 16 bits used in the encryption process are generated from the 128 bit user-selected key as follows: The 128 bit key is partitioned into 8 subblocks that are directly used as the first eight key subblocks (leftmost bit = most significant bit), where the ordering of the key subblocks is defined as follows: $Z_1^{(1)}, Z_2^{(1)}, ..., Z_6^{(1)}, Z_1^{(2)}, ..., Z_6^{(2)}, ..., Z_1^{(8)}, ..., Z_6^{(8)}, Z_1^{(9)}, ..., Z_4^{(9)}$. The 128 bit user-selected key is then cyclic shifted to the left by 25 positions, after which the resulting 128 bit block is again partitioned into eight subblocks that are taken as the next eight key subblocks. The obtained 128 bit block is again cyclic shifted to the left by 25 positions to produce the next eight key subblocks, and this procedure is repeated until all 52 key subblocks have been generated.

3 A partial distributive law

The basic operations used in the design of IDEA are: $\odot$: multiplication modulo $2^{16}+1$ of 16 bit integers, with the zero number corresponding to 2^{16}, $\boxplus$: addition modulo 2^{16} of 16 bit integers and $\oplus$: bit-by-bit exclusive-OR of 16 bit integers. It is stated in [3] that the three operations are *incompatible* in the sense that no pair of these satisfies a distributive law. In this section we show however that the operations $\odot$ and $\boxplus$ satisfy a partial distributive law stemming from arithmetic modulo $2^n + 1$. This law also carries over to a partial arithmetic property of the MA structure in the round function (see Figure 2 in Section 4).
In the ring of integers modulo $2^n + 1$ one has a distributive law, i.e., for any integers a, b, $delta \in \{0, 1, ..., 2^n\}$

$$a \cdot (b + delta) = a \cdot b + a \cdot delta \pmod{2^n + 1} \tag{1}$$

where addition and multiplication are taken modulo $2^n + 1$. Obviously, this law carries over to the operations $\odot$ and $+$:

$$a \odot (b + \mathit{delta}) = a \odot b + a \odot \mathit{delta} \pmod{2^n + 1}. \tag{2}$$

We now ask whether this law even holds modulo 2^n for some fraction of integers a, b, *delta*. For this we compare $a \odot (b \boxplus \mathit{delta})$ with $a \odot b \boxplus a \odot \mathit{delta} \pmod{2^n}$.

Proposition 1. *(1) If $a = 0$ the equation $a \odot (b \boxplus \mathit{delta}) = a \odot b \boxplus a \odot \mathit{delta}$ is satisfied for no b and delta;*
(2) If $a = 1$ the equation $a \odot (b \boxplus \mathit{delta}) = a \odot b \boxplus a \odot \mathit{delta}$ is satisfied for every b and delta;
(3) If $a \neq 0, 1$ the equation $a \odot (b \boxplus \mathit{delta}) = a \odot b \boxplus a \odot \mathit{delta}$ is satisfied for no triple of the form $(a, 0, \mathit{delta})$ or $(a, b, 0)$;
(4) If $a \neq 0, 1$, $b \neq 0$ and $\mathit{delta} \neq 0$, the equation $a \odot (b \boxplus \mathit{delta}) = a \odot b \boxplus a \odot \mathit{delta}$ is satisfied if and only if the two conditions

$$b + \mathit{delta} \leq 2^n$$
$$a \odot b + a \odot \mathit{delta} \leq 2^n$$

are satisfied.

Proof. Statement (1) follows easily by using the fact that $0 \odot b = 1 - b \pmod{2^n}$ for any b. Staments (2) and (3) are trivial. For stament (4) observe that the equation $a \odot (b \boxplus \mathit{delta}) = a \odot b \boxplus a \odot \mathit{delta} \mod 2^n$ holds if $b + \mathit{delta} \leq 2^n$, and $a \odot b + a \odot \mathit{delta} \leq 2^n$. For $a > 1$ also the converse holds: If $b + \mathit{delta} > 2^n$, this leads to a calculation modulo 2^n, giving difference $a \pmod{2^n + 1}$ between $a \odot (b + \mathit{delta})$ and $a \odot (b \boxplus \mathit{delta})$, which cannot be equalized by computing $a \odot b + a \odot \mathit{delta}$ modulo 2^n rather than $2^n + 1$. This also shows that for the equation in (4) to hold, the condition $a \odot b \boxplus a \odot \mathit{delta} \leq 2^n$ is necessary.

If b and *delta* are supposed to be random, one has $b + \mathit{delta} \leq 2^n$ with probability $1/2$ and similarly, $a \odot b + a \odot \mathit{delta} \leq 2^n$ with probability $1/2$.
If a is random, we heuristically assume that these two events may be considered to be independent. This will roughly be justified. Hence, for random a, b and *delta* we may expect the equation in Proposition 1 to hold with probability about $1/4$.

We give first an exact relationship in two cases, where *delta* is fixed (namely the opposite cases $\mathit{delta} = 1$ and $\mathit{delta} = 2^n - 1 = -1 \pmod{2^n}$).

Proposition 2. *Let $n \geq 2$ be an integer. Then for random integers $a, b \in \{0, 1, \ldots, 2^n - 1\}$ the distributive law*

$$a \odot (b \boxplus 1) = a \odot b \boxplus a \quad (\mathit{mod}\ 2^n) \tag{3}$$

holds with probability $1/2 - 2^{-n-1} + 2^{-2n} \approx 1/2$, and the distributive law

$$a \odot (b \boxplus (2^n - 1)) = a \odot b \boxplus a \odot (2^n - 1) \quad (\mathit{mod}\ 2^n) \tag{4}$$

holds with probability $2^{-n} + 2^{-n-1} - 2^{-2n} \approx 2^{-n}$.

Proof. According to Proposition 1 we can suppose $a \geq 1$, and for $a = 1$ the law always holds. As $delta = 1$ for the first part, one has $b + delta = b + 1 \leq 2^n$ for every $b \in \{0, ..., 2^n - 1\}$. Therefore, for every fixed $a > 1$ we count the number of cases, where $a \odot b + a \leq 2^n$, or $a \odot b \leq 2^n - a$. As $a \odot b \pmod{2^n + 1}$ for varying b can take every value between 1 and 2^n, we thus have to exclude a values for b. Hence counting the number of possibilities for $a = 1, 2, ..., 2^n - 1$ we obtain $2^n + (2^n - 2) + (2^n - 3) + ... + 2 + 1$ possibilities in which the law (3) holds. This number is $2^n + (2^n - 1)(2^n - 2)/2 = 2^{2n-1} - 2^{n-1} + 1$. Hence the probability for the law (3) to hold is $(2^{2n-1} - 2^{n-1} + 1) \cdot 2^{-2n} = 1/2 - 2^{-n-1} + 2^{-2n}$ as claimed. If $delta = 2^n - 1$ we again have the case $a = 1$ where the law (4) always holds. For $a > 1$ we have $b + delta = b + (2^n - 1) \leq 2^n$ only for $b = 0$ and $b = 1$. But for $a > 1$ and $b = 0$ the law doesn't hold. So let $b = 1$. Then (4) holds if and only if $a \odot (2^n - 1) = a \odot 2^n - a \pmod{2^n}$. This is true as long as $a \odot 2^n > a$. As $2 \cdot 2^n = 2^{n-1} \pmod{2^n + 1}$, $3 \cdot 2^n = 2^n - 2 \pmod{2^n + 1}$,..., one can see that this is the case for $a = 2, ..., a^{n-1}$. Hence we have $2^n + 2^{n-1} - 1$ possibilities where (4) holds. Therefore the probability is $2^{-n} + 2^{-n-1} - 2^{-2n}$.

Proposition 2 and the considerations made in this section show that, depending on $delta$, the probability for the partial distributive law to hold strongly varies between the values given in (3) and (4), and that this probability decreases for increasing $delta$. But even in the case $delta = 2^n - 1$ with lowest probability, this value is slightly higher than the probability 2^{-n} which one would expect if the validity of the distributive law (mod 2^n) would behave purely randomly. Experiments have shown that the decrease of probability is approximately linear in the increase of $delta$. In fact, for $n = 4$ and $n = 8$, the probability for the partial distributive law to hold for the "average" value $delta = 2^{n-1}$ is extremely near to the arithmetic mean of the two values given in (3) and (4). For $n = 4$ and $n = 8$ these values are 0.28125 and 0.25195, respectively. As these values (especially for $n = 8$) are only slightly higher than 1/4, this further confirms our heuristic considerations, that for random $a, b, delta$ the distributive law holds with probability about 1/4.

3.1 Applications of the partial distributive law

As an immediate consequence of the previous results we give a relationship between $a \odot (b \boxplus delta) - a \odot b$ and $a \odot delta$.
We have $a \odot (b \boxplus delta) = a \odot b \boxplus a \odot delta \pmod{2^n}$ with probability p depending on $delta$ (as explained in Proposition 2). Thus we obtain

$$a \odot (b \boxplus delta) - a \odot b = a \odot delta \pmod{2^n} \tag{5}$$

with the same probability. In particular,

$$a \odot (b \boxplus 1) - a \odot b = a \pmod{2^n} \tag{6}$$

with probability $\approx 1/2$ if n is sufficiently large (e.g., $n = 8$ or $n = 16$).

In view of the IDEA-algorithm, the question will be whether the partial distributive law is of any assistance for cryptanalysis. As a first observation in this direction, we obtain that a key block Z which acts as multiplication $\odot$ can be determined with a certain probability, provided the input and output differences (or sums) to this block are supposed to be known. This probability depends on the input difference and on the magnitude of Z (see Proposition 2 and its proof).
More importantly, consider the distributive law for differences ($a > 0, b > 0$, $c > 0$)

$$a \odot (b - c) = a \odot b - a \odot c \pmod{2^n + 1}. \tag{7}$$

This holds also modulo 2^n as long as $b > c$ and $a \odot b > a \odot c$. Let $delta = b - c > 0$. Then there are $2^n - delta$ pairs $(b, c) = (delta + 1, 1), ..., (2^n, 2^n - delta)$ for which "difference" modulo 2^n and modulo $2^n + 1$ has the same meaning, and ordinary multiplication agrees with $\odot$ modulo $2^n + 1$. If for $a > 0$ we take $a \odot b - a \odot c$ modulo $2^n + 1$ rather than modulo 2^n (and interpreting products $\odot$ as 2^n if they take the value 0), it is the correct difference.
Suppose $Z > 0$ and we know the outputs after multiplication with Z (not only their difference). Take their difference modulo $2^n + 1$. Then, knowing the input difference $delta$ (mod 2^n) we can guess the key block Z with probability $(2^n - delta) \cdot 2^{-n} = 1 - delta \cdot 2^{-n}$. Using considerations as in Proposition 1 we can also detect a subblock $Z = 0$ from knowledge of input and output differences.

Partially arithmetic properties of the MA structure As the MA structure (see Figure (2)) is composed only of multiplications $\odot$ (mod $2^n + 1$) and additions $\boxplus$ (mod 2^n), the partial distributive law carries over to certain relations holding between differences in the inputs p, q and the outputs u, t.

First suppose p is fixed and q_1, q_2 are two different input values with $q_2 = q_1 \boxplus delta$. Then $s_1 = q_1 \boxplus r$ and $s_2 = q_2 \boxplus r = q_1 \boxplus delta \boxplus r = s_1 \boxplus delta$.
Hence $t_2 - t_1 = Z_6 \odot s_2 - Z_6 \odot s_1 = Z_6 \odot (s_1 \boxplus delta) - Z_6 \odot s_1$.
As $Z_6 \odot (s_1 \boxplus delta) = Z_6 \odot s_1 \boxplus Z_6 \odot delta$ with probability P depending on $delta$, we have

$$t_2 - t_1 = Z_6 \odot delta \text{ and } u_2 - u_1 = Z_6 \odot delta \tag{8}$$

with this probability P.
If q is fixed and p_1 and p_2 are two different input values, one gets a more complicated but weaker relationship for the outputs u, t, which we omit to formulate.
In the other direction, suppose the input difference $delta = p_2 - p_1$ and the output differences $t_2 - t_1$ and $u_2 - u_1$ are known. Then the difference $r_2 - r_1 = u_2 - t_2 - (u_1 - t_1)$ can be computed and thus Z_5 can be determined with a probability depending on $delta$. A similar conclusion also holds for Z_6 if both the input differences $p_2 - p_1$ and $q_2 - q_1$ are known.

4 A Class of High-Probability Differentials of IDEA

In [5] various considerations lead to three candidate classes of differentials of potential interest for differential cryptanalysis of the IDEA-algorithm. We give a new derivation of the class having highest probability (under the condition of key independence). Our approach allows for a quite general analytic treatment of these differentials. The arguments give new insight into the interaction of the 3 basic operations and will have consequences in the analysis of few rounds of IDEA in Section 5.

In our discussion we use the notation in [4], [5]. In particular the "difference" ΔX is given by $\Delta X = X \otimes X^{*-1}$, where the operation $\otimes$ is defined on 64 bit blocks by

$$X \otimes X^* = (X_1 \odot X_1^*, X_2 \boxplus X_2^*, X_3 \boxplus X_3^*, X_4 \odot X_4^*) \tag{9}$$

and where X^{*-1} denotes the inverse of X^* under the group operation $\otimes$.

The round function of IDEA is illustrated in Figure 2,where X_i, Y_i denote 16 bit subblocks of the 64 bit plaintext and ciphertext blocks respectively, and $Z_1^{(1)},...,Z_6^{(1)}$ denote 16 bit key subblocks of the first round according to the key scheduling as described in [4], [5]. One can also consider "mini ciphers" where the subblocks are $n = 2$, 4 or 8 bit integers.

For any two n-bit integers a and a^*, write $\delta a = a \odot (a^*)^{-1}$, and $\partial a = a - a^* = a \boxplus (-a^*)$. Then the differences ΔX and ΔY are expressed as $\Delta X = (\delta a, \partial b, \partial c, \delta d)$ and $\Delta Y = (\delta v, \partial w, \partial x, \delta y)$. From Figure 2 one has $(\delta a, \partial b, \partial c, \delta d) = (\delta e, \partial f, \partial g, \delta h)$.

The most probable one-round differentials (thus far known) which may be of use in differential cryptanalysis are of the form

$$(\alpha, \beta) = (1, o_a, 0, 0; 1, 0, o_b, 0) \text{ or } (0, 0, o_a, 1; 0, o_b, 0, 1) \tag{10}$$

Here o_a in the input difference α denotes a (fixed) odd integer between 1 and $2^n - 1$, i.e. $o_a \in \{1, 3, ..., 2^n - 1\}$ and o_b in the output difference β is a (fixed) element of a subset of the odd integers, where this subset is dependent on o_a and will be specified.

This class of differentials is referred to in ([5], Ch. 5) as "differentials based on the trivial transparency of the MA structure". The idea is to choose the input difference α such that the probability $P((\delta p, \partial q) = (1, 0))$ is maximized. This is achieved by fixing , e.g., X_1, X_3 and choosing the difference of the other input blocks appropriately.

In ([5], Ch. 5, Property 7) the values for α were determined by a direct computational search:

For $n = 2, 4, 8$ and 16 and for α of the form $(1, o_a, 0, 0)$ or $(0, 0, o_a, 1)$ where o_a is an odd integer between 1 and $2^n - 1$

$$\begin{aligned} P((\delta p, \partial q) = (1, 0) | \Delta X = \alpha) &= \max_{\sigma} P((\delta p, \partial q) = (1, 0) | \Delta X = \sigma) \\ &= 2^{-(n-1)}. \end{aligned} \tag{11}$$

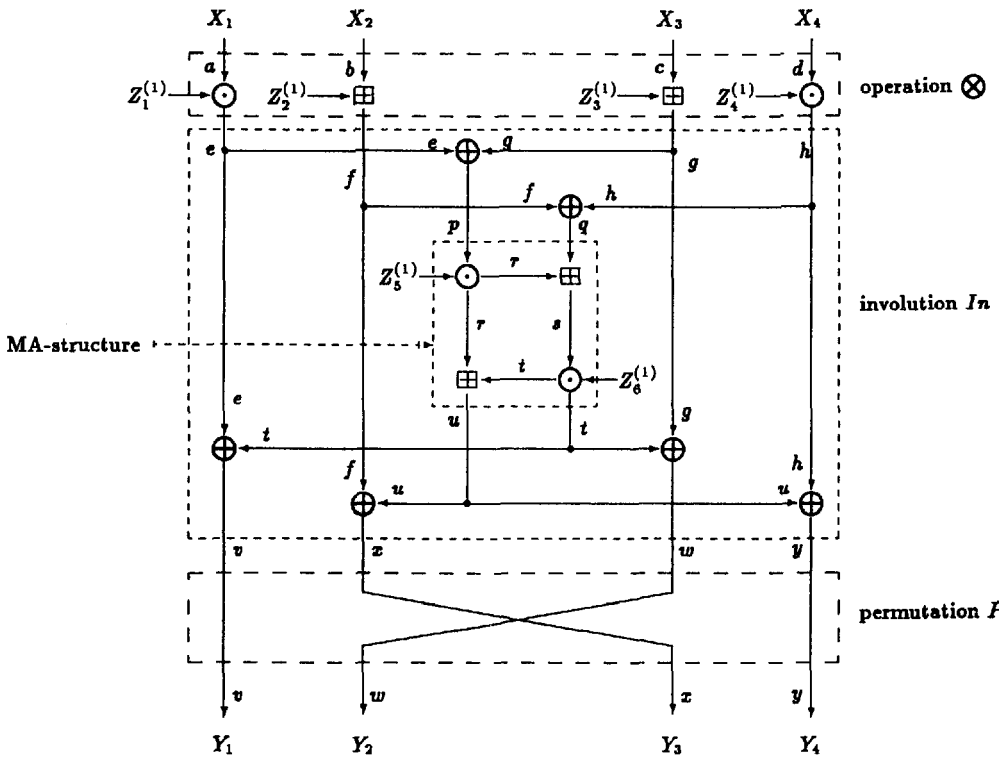


Fig. 2. The round function of IDEA and notation used for analysis.

If we already restrict to input differences with $\delta d = 0$, this computational fact can even be proved analytically for every integer n of interest.

Proposition 3. *Let $n \geq 2$ be an integer such that $2^n + 1$ is prime. Suppose the input difference ΔX is of the form $\Delta X = (1, \partial b, 0, 0)$. Then the probability*

$$P((\delta p, \partial q) = (1,0) | \Delta X = \alpha) \tag{12}$$

is maximized by taking $\alpha = (1, o_a, 0, 0)$, where o_a is any (fixed) odd integer between 1 and $2^n - 1$, and this probability is $2^{-(n-1)}$.

Proof. Denote by $\bar{a}$ the bit-by-bit complement of the n-bit number a. We start with the fact that for any such a

$$\delta a = 0 \Longleftrightarrow \begin{cases} a &= (A, 10...0, \theta) \\ a^* &= (\bar{A}, 10...0, \bar{\theta}) \end{cases}$$

where A is some $[n-(l+1)]$-bit number, $l \epsilon \{0,1,...,n-1\}$, a, a^* contain l consecutive 0's in their binary representation and $\theta \epsilon \{0,1\}$.
Moreover, $\delta a = 0 \Longleftrightarrow a \boxplus a^* = 1$.
For a given difference ∂b we consider all possibilities such that

$$f \oplus h = f^* \oplus h^*. \tag{13}$$

Obviously, ∂b must be an odd number in order the least significant bits in (13) to be equal, so suppose $\partial b = o_a \geq 1$ fixed.
We always get one (positive) solution (f, f^*) with $(f, f^*) = (h, h^*)$ by solving the system of equations

$$\begin{aligned} h + h^* &= 2^n + 1 \\ f - f^* &= o_a \end{aligned} \tag{14}$$

Hence $f = h = 2^{n-1} + (o_a + 1)/2$, $f^* = 2^{n-1} - (o_a - 1)/2$. Every other pair (f, f^*) with condition (13) besides $f - f^* = o_a$ satisfies the conditions

$$f_0 = \bar{f}_0{}^* \tag{15}$$

$$f_i = f_i{}^*, \quad 1 \leq i \leq l+1 \tag{16}$$

$$f_i = \bar{f}_i{}^*, \quad i > l+1 \tag{17}$$

Here l is the number of consecutive 0's in the binary representation of h and f_i denotes the i-th bit in the binary representation of f. We show that the number l is uniquely determined by the difference o_a. Suppose, on the contrary, that we have pairs (f, f^*), and (k, k^*) both having difference o_a and satisfying conditions (15),(16),(17) for integers l and l', respectively, with $l' < l$, say. Then already $k_i = \bar{k}_i{}^*$ for $l+1 \geq i > l'+1$. Compared to the bits in (f, f^*),these bits necessarily contribute to a change in the difference $k - k^*$ of the form $2^{l'+2} \cdot o$, where o is an odd integer. But this change cannot be compensated by simultaneously complementing (some or all) bits k_i, k_i^* for $i > l+1$ or $i = 0$, as such a change would be of the form $2^j \cdot o_1 + \epsilon, \epsilon = 0$ or ± 2, where o_1 is odd and where $j \geq l+2 > l'+2$. As a consequence we have

- The number l can be determined out of the system of equations (14)
- One gets all possibilities for (f, f^*) out of the solution of (14) by either simultaneously complementing some or all bits in f and f^* which agree, i.e., for $1 \leq i \leq l+1$, or the most significant bit.

As the number of equal bits in (f, f^*) is $l+1$, the number of possibilities for f is 2^{l+2}, or the probability to get an f of this form is 2^{-n+l+2}. The probability to get a h fitting this l is 2^{-l-1}. Hence, if we choose b and d (or f and h) randomly, the probability for $f \oplus h = f^* \oplus h^*$ to hold is $2^{-n+l+2} \cdot 2^{-l-1} = 2^{-n+1}$. Note that this probability is independent of the initially chosen o_a.

As a consequence of Proposition 3 we get

Proposition 4. *Let $n \geq 2$ be an integer such that $2^n + 1$ is prime. Suppose the input difference ΔX is of the form $\Delta X = (1, o_a, 0, 0)$ where o_a is a fixed odd integer between 1 and $2^n - 1$. Then*

$$P((\delta v, \partial w) = (1,0)) = 2^{-n+1} \tag{18}$$

Proof. $\partial w = 0$ implies $\partial t = 0$, as $\partial g = 0$. Hence $\partial q = 0$ as $\partial s = 0$ and $\partial r = 0$. This means that $(\delta v, \partial w) = (1,0) \Longleftrightarrow (\delta p, \partial q) = (1,0)$. Therefore Proposition 3 applies.

In order to get also a statement for ∂x and δy, we compare $f \oplus u$ with $f^* \oplus u$ and similarly, $h \oplus u$ with $h^* \oplus u$.
As observed in the proof of Proposition 3 the chosen difference o_a determines the number l of consecutive 0's in h in order to satisfy the equation $f \oplus h = f^* \oplus h^*$. We have $\delta y = 0$ exactly if all these consecutive 0's together with the subsequent bit 1 remain unchanged after XORing with u. This means that $u_i = 0$ for $1 \leq i \leq l+1$. As u takes its values in $\{0, 1, 2, ..., 2^n - 1\}$ uniformly and independently of f and h, this event happens with probability 2^{-l-1}.
On the other hand, the difference o_a may change into a difference ∂x by XORing f with u. A change by ± 2 happens if the least significant bit is complemented. Moreover, a change necessarily also happens, if some of the bits f_i, $l+1 < i < n$, are complemented. Note that the change in difference is always an even number. Therefore the output difference is always an odd number o_b. The probability that $f \oplus u - f^* \oplus u = o_a$, i.e., that $u_i = 0$ for $i = 0$ and $l + 1 < i < n - 1$, is 2^{-n+l+1}. As the bits of u causing changes in ∂x and δy are disjoint *except* u_{n-1}, the differences ∂x and δy remain unchanged *simultaneously* iff $u = 0$ or $u = 2^{n-1}$. The probability for this event is 2^{-n+1}.
Depending on o_a (and therefore on l), output differences $\Delta Y = (1, 0, o_b, 0)$ for o_b in a restricted set of odd numbers are possible. Hereby all differences ΔY for a realizable o_b have the same probability. In particular, $o_b = o_a$ is always realizable.
Thus our considerations together with Proposition 4 prove the following result:

Theorem 5. *Let $n \geq 2$ be an integer such that $2^n + 1$ is a prime number and let o_a denote an arbitrary odd integer between 1 and $2^n - 1$. Then*

$$P(\Delta Y = (1, 0, o_a, 0) | \Delta X = (1, o_a, 0, 0)) = 2^{-2(n-1)}. \tag{19}$$

This extends a result in ([5], Ch. 5) from $o_a = 1$ to arbitrary odd integers.

Theorem 5 deserves some remarks:
Instead of differentials $(\alpha, \beta) = (1, o_a, 0, 0; 1, 0, o_b, 0)$ one can also consider differentials of the form $(\alpha, \beta) = (0, 0, o_a, 1; 0, o_b, 0, 1)$. Analytic considerations as well as experiments show that a result equivalent to Theorem 5 also holds for this class of differentials.

In Proposition 4 we have seen that probabilities for output differences restricted to the first two output subblocks are much higher than the probability given in Theorem 5 for differences considered over all output subblocks simultaneously. Our considerations leading to Theorem 5 show that depending on the integer l, and therefore on the input difference o_a, a similar statement is still true, if we restrict to differences in only three of the four output subblocks:

$$P((\delta v, \partial w, \partial x) = (1,0,o_a)|\Delta X = (1,o_a,0,0)) = 2^{-n+1} \cdot 2^{-n+l+1} = 2^{-2(n-1)+l} \quad (20)$$

$$P((\delta v, \partial w, \delta y) = (1,0,0)|\Delta X = (1,o_a,0,0)) = 2^{-n+1} \cdot 2^{-l-1} = 2^{-n-l} \quad (21)$$

Here l is the number determined by equations (14) for given o_a, e.g., for $o_a = 1$ one has $l = n-2$, and hence these two probabilities are 2^{-n} and $2^{-2(n-1)}$ respectively. The other extreme case is $l = 0$, which occurs, e.g., if $o_a = 3$. The probabilities in (20) and (21) are then $2^{-2(n-1)}$ and 2^{-n}, respectively.

4.1 Information on Subkeys for Known Input-Output Differences

The probability derived in Theorem 5 for the class of differentials $(\alpha,\beta) = (1,o_a,0,0;1,0,o_b,0)$ is independent of the subkeys $Z_1^{(1)},...,Z_6^{(1)}$ of one round. Nevertheless, the known occurence of (part of) such a differential for a known plaintext pair (X,X^*) with difference α allows to derive considerable information on the subkeys $Z_1^{(1)}, Z_2^{(1)}, Z_3^{(1)}$ and $Z_4^{(1)}$.

Suppose a plaintext pair (X,X^*) with difference $\Delta X = (1,o_a,0,0)$ is submitted and produces the known (or anticipated) output difference $\delta v = 1$ (or equivalently $\partial w = 0$) after one round. Hence $t = t^*$. As $p = p^*$ this implies $u = u^*$ as $r = r^*$, and hence $q = q^*$. So we know that with this pair (X,X^*) of plaintexts the event $h \oplus f = h^* \oplus f^*$ has occured. (According to Proposition 4 for IDEA we have to make 2^{15} trials in the average until this event occurs). Suppose now $o_a = 1$. Then the equations (14) give $l = n-2$, so there remain only four possibilities for h, namely $0 = (0,...,0)$, $1 = (0,...,1)$, $2^{15} = (1,0,...,0)$ and $2^{15}+1 = (1,0,...0,1)$. Hence for known X there remain only four possibilities for $Z_4^{(1)}$. (In addition, the least significant bit of $Z_2^{(1)}$ is determined). Similar (but slightly weaker) conclusions can also be drawn on the subkey $Z_1^{(1)}$ for differentials of type $(\alpha,\beta) = (0,0,o_a,1;0,o_b,0,1)$.

On the opposite side, suppose o_a is such that the equations (14) give $l = 0$. This is the case, e.g., for $o_a = 3$. Then there remain only four possibilities for f, namely $2^{n-1}+(o_a+1)/2$, $2^{n-1}-(o_a-1)/2$, $(o_a+1)/2$ and $-(o_a-1)/2$. Thus, if X is supposed to be known, there remain only four possibilities for $Z_2^{(1)}$.

5 Analysis of IDEA with a reduced number of rounds

The aim of this section is to give estimates for the computational complexity to break the first few rounds of IDEA by combining known results as well as the arithmetic properties we have found in previous sections. In our discussion

we are unable to break more than 2 rounds of IDEA but a (rough) estimate indicates the number of rounds that are at least needed so that a complete exhaustive search will be necessary in order to find the secret key. We outline first a proposal how to break a 2-round IDEA.

2-round IDEA: The problem of analysing a 2-round IDEA is split up into determining the subkey blocks $Z_5^{(2)}$ and $Z_6^{(2)}$ (in the MA structure of the second round) under the assumption that the output $(Y_1^{(2)},...,Y_4^{(2)})$ after the second round (without permutation P_I and without "output transformation") is known, and then analysing a "one and a half" round IDEA (i.e., the first round with output transformation).

The idea is to make a search over $Z_5^{(2)}$, $Z_6^{(2)}$, by composing the 2-round IDEA block cipher, denoted by $F(\mathbf{x},\mathbf{k})$, with the involution In (see Figure 2) with a chosen pair $(Z_5^{(2)}, Z_6^{(2)})$ of key subblocks. Note that this composition $In \circ F$ agrees with the one and a half round IDEA provided we have found the correct pair $(Z_5^{(2)}, Z_6^{(2)})$. We further observe that the partial distributive law applied to the MA structure is of limited use to determine $Z_5^{(2)}$, $Z_6^{(2)}$, as this would need *simultaneous* knowledge of the differences $t^{(2)} - t^{(2)*}$ and $u^{(2)} - u^{(2)*}$ in the second round. But the final XOR's in the involution In leave many choices in general for these differences, even with some knowledge of input differences to In (e.g., using differentials in the first round). Therefore we make an exhaustive search over $(Z_5^{(2)}, Z_6^{(2)})$. A choice $(Z_5^{(2)}, Z_6^{(2)})$ could be tested for correctness by choosing plaintexts X, X^* with $\Delta X = (1,1,0,0)$, according to the differentials studied in Section 4. A faster method appears to be based on a consideration in [5] and essentially going back to [4], namely that for $n = 16$

$$P(\delta v = 1, \delta w = 0, \delta x = 0, \delta y = 2^{16} - 1 | \Delta X = (0,1,0,0)) \approx 2^{-9}. \qquad (22)$$

Thus we choose plaintexts X, X^* with $\Delta X = (0,1,0,0)$. Then by (22) and by the (refined) partial distributive law applied to the key subblock $Z_4^{(2)}$ and with negative input difference $-\delta y = 1$ we have, at the beginning of the second round (using notation similar as in Figure 2),

$$P(\delta e^{(2)} = 1, f^{(2)} \boxplus f^{(2)*} = 2Z_2^{(2)} \boxplus 1, g^{(2)} \boxplus g^{(2)*} = 2Z_3^{(2)} \boxplus 1, \\ h^{(2)*} - h^{(2)} = Z_4^{(2)}) \approx 2^{-9}. \qquad (23)$$

Although the (constant) key subblocks in (23) are unknown, and (22) is not a differential for IDEA in terms of difference as defined by (9), we may still use relation (23) as a test whether we have found the correct pair $(Z_5^{(2)}, Z_6^{(2)})$ in an exhaustive search. This is based on the (unproven but plausible) hypothesis that for chosen plaintexts X, X^* with $\Delta X = (0,1,0,0)$ the outputs (v,x,w,y) and (v^*,x^*,w^*,y^*) of the cipher $In \circ F$ satisfy:

If the MA structure in the involution In has been loaded with the correct pair $(Z_5^{(2)}, Z_6^{(2)})$ of key subblocks there exist odd integer numbers o_x, o_w and an integer c_y (which in general are not unique) such that the probability

$$P(\delta v = 1, x \boxplus x^* = o_x, w \boxplus w^* = o_w, y^* - y = c_y) \qquad (24)$$

is significantly higher than the corresponding probability for most other (incorrect) pairs of key subblocks.

This hypothesis has been tested and verified experimentally in the case of the IDEA mini-cipher with $n = 4$. The ambiguity of the constants o_x, o_w and c_y is due to the (experimental) fact that there exist key-dependent one-round differentials of high probability. However, our experiments suggest, that for most keys the cipher $In \circ F$ has no such differentials.

Informally, a pair $(Z_5^{(2)}, Z_6^{(2)})$ is accepted to be correct if for this pair the corresponding expressions in the output subblocks of the composition $In \circ F$ satisfy (24) for suitable 16 bit integer constants. According to (23) the computational complexity of this search is roughly of magnitude $2 \cdot 2^9 \cdot 2^{32} = 2^{42}$, which is on the verge of practical feasibility. We are thus reduced to find the other key blocks by breaking (part of) the one and a half rounds of IDEA. According to Section 4.1 the number of possible subkeys $Z_1^{(1)}, ..., Z_4^{(1)}$ can be reduced to 256 possibilities in less than 2^{20} trials. In order to determine the other key subblocks of the first round we make explicit use of the key scheduling. Recall that the 128 bit user-selected key is partitioned into 8 subblocks that are directly used as the first eight key subblocks, where the ordering of the key subblocks is defined as follows: $Z_1^{(1)}, Z_2^{(1)}, ..., Z_6^{(1)}, Z_1^{(2)}, ..., Z_6^{(2)}, ...$ The 128 bit user-selected key is then cyclic shifted to the left by 25 positions to give the next 8 key subblocks, and so on.

Suppose the key subblocks $Z_5^{(2)}, Z_6^{(2)}$ have been determined by the procedure as described above. Then according to the key scheduling $Z_5^{(2)}$ agrees with the last 7 bits of $Z_4^{(1)}$ (which may reduce the uncertainty in the previous estimate of $Z_4^{(1)}$) and the first 9 bits of $Z_5^{(1)}$. Similarly, $Z_6^{(2)}$ determines the last 7 bits of $Z_5^{(1)}$ and the first 9 bits of $Z_6^{(1)}$. We complete our knowledge of the remaining key subblocks entering the one and a half rounds of IDEA by a search over the 7 unknown bits of $Z_6^{(1)}$. For this we choose one of the remaining 256 (or less) possibilities of the quadruple $(Z_1^{(1)}, ..., Z_4^{(1)})$. Then every choice of the last 7 bits of $Z_6^{(1)}$ for given input determines (v, w, x, y) and thus $(Z_1^{(2)}, ..., Z_4^{(2)})$, as the output is supposed to be known. Hence the actual choice of the eight key subblocks can now be found by at most $2^7 \cdot 2^8 = 2^{15}$ trials. This shows that the previous search for the pair $(Z_5^{(2)}, Z_6^{(2)})$ is more time consuming than breaking one and a half rounds of IDEA. Hence an optimistic estimate (from the point of view of a cryptanalyst!) predicts about 2^{42} trials to be necessary for breaking the first two rounds of IDEA.

r-round IDEA, $r \geq 3$: For the estimation of the computational complexity of more than 2 rounds we first note that to date no key-independent 2-round differentials with high probability have been found (see [5]). This has also been confirmed by experiments with a mini-IDEA for $n = 4$. Moreover we have found no arithmetic property that might facilitate breaking more than two rounds. Thus to find the subkeys $Z_1^{(3)}, ..., Z_4^{(3)}$ we have no better method than exhaustive search. Therefore, the computational amount to break a two and a half round IDEA is at least $2^{42} \cdot 2^{64} = 2^{106}$.

Proceeding further, breaking a 3-round IDEA needs a full exhaustive search. This suggests that the newly found arithmetic properties for random keys give no advantage in the cryptanalysis of the full IDEA block cipher containing 8 rounds. However these properties show the importance of the fact that in the design of IDEA *three different* group operations have been chosen.

Acknowledgements

I am grateful to X. Lai and J. L. Massey (ETH Zürich) and Th. Brüggemann, H. Bürk, K. Messerli, J.-M. Piveteau and D. Profos (Ascom Tech AG) for many helpful discussions and for their support of this work.

References

1. E. Biham, A. Shamir, Differential Cryptanalysis of DES-like Cryptosystems, Journal of Cryptology, Vol.4, No. 1, 1991, pp. 3-72.
2. Th. Brüggemann, H. Bürk, Der Verschlüsselungsalgorithmus $IDEA^{TM}$, Elektronik, Heft 10, Francis-Verlag, 1993.
3. X. Lai and J. L. Massey, A Proposal for a New Block Encryption Standard, Advances in Cryptology-EUROCRYPT'90, Proceedings, Lecture Notes in Computer Science, Springer-Verlag, Berlin 1991, pp. 389-404.
4. X. Lai, J. L. Massey and S. Murphy, Markov Ciphers and Differential Cryptanalysis, Advances in Cryptology - EUROCRYPT'91, Proceedings, Lecture Notes in Computer Science, Springer-Verlag, Berlin 1991, pp. 17-38.
5. X. Lai, On the Design and Security of Block Ciphers, ETH Series in Information Processing, Editor: J. L. Massey, Vol. 1, 1992.