Cryptanalysis of CTC2

Orr Dunkelman[1,⋆] and Nathan Keller[2,⋆⋆]

[1] Département d'Informatique,
École normale supérieure
45 rue d'Ulm,
Cedex 05, Paris 75230, France
orr.dunkelman@ens.fr
[2] Einstein Institute of Mathematics,
Hebrew University
Jerusalem 91904, Israel
nkeller@math.huji.ac.il

Abstract. CTC is a toy cipher designed in order to assess the strength of algebraic attacks. While the structure of CTC is deliberately weak with respect to algebraic attacks, it was claimed by the designers that CTC is secure with respect to statistical attacks, such as differential and linear cryptanalysis. After a linear attack on CTC was presented, the cipher's linear transformation was tweaked to offer more diffusion, and specifically to prevent the existence of 1-bit to 1-bit approximations (and differentials) through the linear transformation. The new cipher was named CTC2, and was analyzed by the designers using algebraic techniques.

In this paper we analyze the security of CTC2 with respect to differential and differential-linear attacks. The data complexities of our best attacks on 6-round, 7-round, and 8-round variants of CTC2 are 64, 2^{15}, and 2^{37} chosen plaintexts, respectively, and the time complexities are dominated by the time required to encrypt the data.

Our findings show that the diffusion of CTC2 is relatively low, and hence variants of the cipher with a small number of rounds are relatively weak, which may explain (to some extent) the success of the algebraic attacks on these variants.

1 Introduction

The merits of algebraic attacks [8, 13] are still debated in the cryptographic community. In the field of stream ciphers, algebraic attacks succeeded significantly, allowing to break several ciphers much faster than other known techniques (see [7, 12]). At the same time, in the field of block ciphers, the situation is more complicated: while there are several instances in which algebraic attacks can be applied (see [5, 11, 16]), algebraic attacks did not succeed in attacking any

[⋆] This author is supported by the France Telecom Chaire.

[⋆⋆] This author is supported by the Adams Fellowship Program of the Israel Academy of Sciences and Humanities.

well known block cipher better than other techniques. Hence, the applicability of algebraic attacks to block ciphers have stirred quite a debate in the cryptographic community, arguing whether this class of attacks can indeed impose a threat to “strong” block ciphers.

In [8], Courtois presented the block cipher CTC, intended to provide an example of a “cryptographically strong” cipher that can be attacked using algebraic techniques. CTC is an SP network, with scalable parameters. The most interesting set of parameters is when the block and key sizes are of 255 bits, and a large number of 3-bit S-boxes (its six-round variant was broken using algebraic approaches). The components of the cipher are intentionally weak with respect to algebraic attacks: the S-boxes are 3-bit to 3-bit, and the linear transformation is extremely simple. It was claimed in [8] that despite these weaknesses, the cipher is strong against statistical attacks, such as differential and linear cryptanalysis.

The resistance of CTC against linear cryptanalysis was challenged in [14]. It was shown that the linear transformation of CTC, along with the particular S-box used in the cipher, allow for an iterative linear approximation with a single active S-box in every round. This approximation has a bias of $1/4$ per round, thus allowing to mount a simple key recovery attack on r-round CTC with data complexity of about 2^{2r+2} known plaintexts. Moreover, it was shown that for a large portion of the 3-bit S-boxes, similar results would hold.

As a response to the suggested attack, CTC’s designers changed the linear transformation of the cipher and introduced CTC2 [9, 10]. The new linear transformation no longer allows for one bit to depend only on one bit. Despite this fact, the diffusion of the new linear transformation is still very weak as most of the bits after the linear transformation depend on only two bits before the linear transformation (except for one bit which depends on three bits). The weak diffusion can be combined with the 3-bit to 3-bit S-boxes to construct high probability differentials and linear approximations for a small number of rounds.

In this paper we examine the security of CTC2 against statistical attacks. We start by examining 3-round and 4-round variants of the cipher. We present a meet-in-the-middle attack on 4-round CTC2 with data complexity of 4 chosen plaintexts and a negligible time complexity. Furthermore, we present many “structural” differential characteristics (with probability 2^{-14} each) and linear approximations (with bias 2^{-8} each) for 3-round CTC2, and also a 4-round differential with probability 2^{-18} and a 4-round linear approximation with bias 2^{-10}.

We then present truncated differential attacks against 5-round, 6-round, and 7-round variants of CTC2. The data complexities of the attacks are 24, 64, and 2^{15} chosen plaintexts, respectively, and the time complexities are dominated by the time required for encrypting the data.

Finally, we present differential-linear attacks against 7-round and 8-round variants of the cipher. The 8-round attack has data complexity of 2^{37} chosen plaintexts and time complexity of 2^{37} encryptions.

It is important to note that our attacks do not imply that algebraic attacks on block ciphers are unfit to play a role in the cryptanalytic toolbox. However, it seems that our findings show that the diffusion in CTC2 is insufficient to offer security when the number of rounds is relatively small (e.g., 8 or less), which may explain the success of the algebraic attack on CTC2 with a small number of rounds.

The rest of the paper is organized as follows: Section 2 shortly describes the CTC and CTC2 block ciphers. In Section 3 we present an efficient meet-in-the-middle attack on 4-round CTC2. We discuss short differentials and linear approximations of CTC2 in Section 4. In Section 5 these differentials are used to present truncated differential attacks on CTC2, and in Section 6 the differentials and the linear approximations are combined in differential-linear attacks on CTC2. We discuss our results in Section 7.

2 A Description of CTC2

CTC is a toy cipher presented for the sake of cryptanalysis using algebraic attacks [8]. The cipher supports a variable block size, and a variable number of rounds. After an iterative linear approximation with a large bias was presented in [14], the designers of CTC introduced a tweaked version of the cipher called CTC2 [9, 10].

The most discussed version of CTC has a block size and a key size of 255 bits. Each round is composed of an XOR with a subkey, parallel application of the same S-box, and a simple linear transformation. After the last round another subkey is XORed to the output.

The 3-bit to 3-bit S-box used in CTC is $S[\cdot] = \{7, 6, 0, 4, 2, 5, 1, 3\}$. The state is initialized to the plaintext XORed with the first subkey, and the bits enter the S-boxes in groups of three consecutive bits, where bit 0 is the least significant bit of the first S-box (which we denote by S-box 0) and bit 254 is the most significant bit of the 85th S-box (which we denote by S-box 84).

The linear transformation of CTC is very simple, and each output bit, denoted by Z_i, depends on one or two of the input bits, denoted by Y_i. We note that in [8] the notations are slightly different (as they include the round number before the number of the bit):

$$\begin{cases} Z_2 = Y_0 \\ Z_{i \cdot 202+2 \bmod 255} = Y_i \oplus Y_{i+137 \bmod 255} \text{ for } i = 1, \ldots 254 \end{cases}$$

The key schedule of CTC is also very simple: The ith round subkey is obtained from the secret key by a cyclical rotation to the left by i bits.

The main difference between CTC and CTC2 is a different linear transformation used in CTC2 [9, 10]:

$$\begin{cases} Z_{151} = Y_2 \oplus Y_{139} \oplus Y_{21} \\ Z_{i \cdot 202+2 \bmod 255} = Y_i \oplus Y_{i+137 \bmod 255} \text{ for } i = 0, 1, 3, 4 \ldots 254 \end{cases}$$

In addition, the key schedule of CTC2 is slightly different: The ith round subkey is obtained from the secret key by a *right* rotation by i bits (instead of a left rotation used in CTC).

3 A Meet-in-the-Middle Approach

The relatively slow diffusion of CTC2 allows to mount simple meet-in-the-middle attacks on small number of rounds of the cipher. As an example, we present a 4-round attack following the methodology presented in [15]. The attack exploits the fact that there are bits in the intermediate state after three full rounds of encryption, that depend on less than 255 plaintext bits. Consider a concrete bit x having this property, and denote the set of plaintext bits x depends on by S_1. It is clear that if two plaintexts have the same value in all the bits of S_1, then the corresponding intermediate values after three rounds agree on the bit x.

To attack 4-round CTC2, we consider several plaintexts having the same value in the bits of S_1. We would like to partially decrypt the ciphertexts through the last round and check whether the intermediate values after three rounds agree on the bit x. Since the fourth round is the last one, we can swap the order of the linear transformation and the key addition in the fourth round, and hence guessing three bits of an equivalent subkey is sufficient to recover the value of the bit x.[1] For each guess of these three bits, we check whether the suggested values of the bit x are the same for all the encryptions, and if not, we discard the key guess. The right key guess is expected to pass this filtering, and the probability that at most one wrong key guess passes the filtering is as high as 78%, if 4 plaintexts are used in the attack (for 6 plaintexts, only the right subkey remains with probability 80%). Hence, the attacker gains two bits of key information. The time complexity of the attack is dominated by the partial decryption of the first two ciphertexts, since after them the number of candidate subkeys is smaller. By using a simple precomputed table, the attacker can retrieve the possible subkey values by a simple table query.

We note that while the attack seems to retrieve only a small number of subkey bits, it is possible to obtain additional key information by repeating the attack with a different bit instead of x. This would result in an increase in the data complexity of the attack, but this increase can be moderated by using structures, allowing to re-use the same data in attacks with different values of the bit x. For example, the sets of unaffected plaintext bits corresponding to the two least significant bits (i.e., $x = e_0$ and $x = e_1$)[2], share seven joint bits, and hence the attacks with $x = e_0$ and $x = e_1$ can be mounted simultaneously using the same data set.

3.1 Attacking More Rounds Using the 4-Round Meet-in-the-Middle Attack

It is possible to use the above attack to speed-up exhaustive key search on more than 4 rounds of CTC2 when only four chosen plaintexts are available

[1] This standard technique is used in all the attacks presented in this paper. In the sequel, we do not mention it explicitly, and use the term "equivalent subkey", referring to the subkey resulting from the swap.

[2] Throughout the paper e_i denotes a value of 0's in all positions but position i. Similarly, $e_{i,j} = e_i \oplus e_j$, $e_{i,j,k} = e_{i,j} \oplus e_k$, etc.

to the attacker. The attack is based on guessing the subkey of the last rounds, partially decrypting the ciphertexts, and then checking whether the meet-in-the-middle attack "succeeds" (i.e., retrieves a subkey which agrees with the partial decryption).

To attack r-round CTC2, for each key guess, the attacker has to decrypt two of the ciphertexts (which takes time proportional to $2 \cdot (r-4)$ rounds in a trivial implementation). Then, the attacker checks whether the subkey the two ciphertexts suggest agrees with the key guess. In case the check succeeds, then the attacker partially decrypts the remaining pairs, and checks them. Only if the 4-round attack succeeds with the four plaintext/ciphertext pairs, then the attacker performs the full trial encryption. As the first step is the most time consuming, the time complexity of this approach is about $r/(2 \cdot (r-4))$ times faster than exhaustive key search (about 60% speed-up for a 5-round attack, a 33% speed-up for a 6-round attack, and a 14% speed-up for a 7-round attack).

Moreover, the attacker can use a slightly better implementation of the partial decryptions (which are done under the same subkeys). For example, by implementing the partial decryption step in a bit-slice manner (which is expected to be the faster implementation of CTC2's decryption in any case), and using the fact that most modern CPUs can support several operations in parallel (or at least have higher throughput by scheduling the operations correctly), one can reduce the time required for the decryption of two values to almost the time required for decrypting one value. In such a case, the expected speed-up is going to be about $r/(r-4)$, i.e., offer a 80% speed-up for a 5-round attack and a 66% speed-up for a 6-round attack.

4 Linear Approximations and Differentials of CTC2

4.1 Linear Approximations

Due to the low diffusion of CTC2, there are many linear approximations for a small number of rounds with a relatively high bias. These approximations exploit the following two properties of CTC2:

1. The S-box of CTC2 has several one-bit to one-bit linear approximations. We note that this property is not a weakness of the particular S-box used in CTC2. The majority of the 3-bit to 3-bit S-boxes have such approximations.
2. The diffusion of the linear transformation in the backward direction is extremely low: Each bit after the linear transformation, except for one, is the XOR of only two bits before the linear transformation. The remaining bit is the XOR of three bits.

Using these properties, it is easy to construct many 3-round linear approximations having only seven active S-boxes: four active S-boxes in the first round, two active S-boxes in the second round, and one active S-box in the third round. All these approximations have a bias of $\pm 1/4$ for each active S-box, or a total of $q = \pm 2^6 \cdot (2^{-2})^7 = \pm 2^{-8}$. Most of them can be easily extended by one (or two) more rounds in the backward direction, where in the additional rounds there are eight (or 24, respectively) more active S-boxes. The following is an example of such approximation:

$$\begin{aligned} \lambda_P = e_{14,104,134,241} &\xrightarrow{S} e_{14,104,132,241} & p = \tfrac{1}{2} - 2^{-5} \\ \xrightarrow{LT} e_{38,154} &\xrightarrow{S} e_{36,154} & p = \tfrac{1}{2} + 2^{-3} \\ \xrightarrow{LT} e_0 &\xrightarrow{S} e_2 = \lambda_C & p = \tfrac{1}{2} + 2^{-2} \end{aligned}$$

We note that there are many ways to change the approximation slightly and get similar approximations with bias 2^{-8} and the same output mask. Hence, attacks which exploit multiple linear approximations [4, 6] are expected to succeed against CTC2 significantly better than a standard linear attack.

While the diffusion of the linear transformation in the backward direction is very low, the diffusion in the forward direction is extremely high.[3] Most of the bits before the linear transformation are the XOR of more than 50 bits after the linear transformation, and hence most of the linear approximations presented above cannot be extended in the forward direction. However, there exist several special bits whose linear diffusion in the forward direction is weak. In particular, bit 2 before the linear transformation is the XOR of bits 30 and 151 after the linear transformation. As a result, the specific approximation presented above can be extended by one more round by concatenating the following 1-round approximation:

$$e_2 \xrightarrow{LT} e_{30,151} \xrightarrow{S} e_{32,151} = \lambda_C \qquad p = \tfrac{1}{2} - 2^{-3}$$

The resulting 4-round approximation (with bias 2^{-10}) cannot be extended further due to the high diffusion of bits 32 and 151 in the forward direction.

In total, the best linear approximations we detected for three, four and five rounds of the cipher have biases of $2^{-6}, 2^{-10}$, and 2^{-18}, respectively.

4.2 Differential Characteristics

The analysis of short differentials of CTC2 is similar to the analysis of short linear approximations presented above. The only difference is that for differentials, the diffusion in the *forward* direction is very weak (i.e., a difference in a single bit before the linear transformation influences only two or three bits after the linear transformation), while the diffusion in the *backward* direction is extremely high.[4] As a result, there are many 3-round differentials with only seven active S-boxes, that can be extended in the forward direction by one or two more rounds

[3] We note that this situation is not typical in block ciphers. In most of the ciphers, the diffusions in both directions are comparable, and hence usually approximations have a round with a single active S-box in the middle, and then are extended to both directions. In CTC2, the round with a single active S-box can be placed only at the end of the approximation.

[4] This situation is quite general in block ciphers. Differentials in the forward direction usually correspond to linear approximations in the backward direction and vice versa, see [1].

(with 8 or 24 more active S-boxes, respectively). One of these differentials is the following:

$$\begin{array}{rl} \Omega_P = e_2 \xrightarrow{S} e_0 & p = 2^{-2} \\ \xrightarrow{LT} e_{2,123} \xrightarrow{S} e_{0,123} & p = 2^{-4} \\ \xrightarrow{LT} e_{2,113,123,234} \xrightarrow{S} e_{0,111,123,234} = \Omega_C & p = 2^{-8} \end{array}$$

Due to the low diffusion of bit 2 in the backward direction, this specific differential can be extended in the backward direction by adding the following one-round differential:

$$e_{30,150} \xrightarrow{S} e_{30,151} \xrightarrow{LT} e_2 \qquad p = 2^{-4}$$

The resulting four-round differential has probability 2^{-18}. In total, the best differential characteristics we detected for $3, 4$, and 5 rounds of the cipher have probabilities $2^{-14}, 2^{-18}$, and 2^{-34}, respectively.

5 Differential Attacks on CTC2

5.1 A 5-Round Differential Attack

The differentials presented above can be used to construct a high-probability truncated differential for 5 rounds of CTC2. The truncated differential starts with the first two rounds of the above 4-round differential:

$$\begin{array}{rl} \Omega_P = e_{30,150} \xrightarrow{S} e_{30,151} & p = 2^{-4} \\ \xrightarrow{LT} e_2 \xrightarrow{S} e_0 & p = 2^{-2} \end{array}$$

From e_0 (before the linear transformation of the second round), the difference propagates unconstrained for 3 more rounds. If the two first rounds of the differential hold, then there is no difference in 35 bits after the fifth round. Denote the set of these 35 bits by S_2.[5]

This truncated differential can be used to mount a simple attack on 5-round CTC2. The attacker considers 64 pairs of plaintexts with difference Ω_P, and checks whether in the corresponding ciphertext pairs, the difference in the bits of S_2 is zero. Since for a random pair, the probability that a difference in 35 bits is zero is 2^{-35}, it is expected that only pairs satisfying the differential (the right pairs) pass this filtering. The probability of the differential is $1/64$, and hence it is expected that only the right pair remains after the filtering. Once identified, this pair can be used to retrieve the subkey used in S-boxes 10 and 50 in the first round.

[5] For completeness we give the set S_2:

$$S_2 = \{13, 14, 24, 35, 50, 56, 59, 65, 66, 74, 79, 80, 82, 112, 116, 118, 131, 132, 135, 148, 165, 169, 171, 184, 187, 190, 199, 200, 201, 222, 226, 237, 240, 243, 252\}.$$

The data complexity of the attack can be reduced using structures. The attacker considers a structure of 24 plaintexts, in which the value in all the S-boxes except for S-boxes 10 and 50 is constant, and the values in these two S-boxes are distinct. These plaintexts can be combined into $24 \cdot 23/2 = 276$ pairs, and about 4 of them[6] are expected to have difference only in bit 2 after the first round. Since the probability of the second round of the differential is 2^{-2}, it is expected that the data contains one right pair. As in the basic attack, this right pair can be detected immediately (by checking whether the ciphertexts have zero difference in the bits of S_2) and used to retrieve the subkeys used in S-boxes 10 and 50 in the first round. The identification of the right pair is immediate when using a hash table, and the complexity of the attack is dominated by the encryptions of the 24 plaintexts.

5.2 A 6-Round Differential Attack

Using the low diffusion of CTC2, the truncated differential presented above can be used to attack 6-round CTC2.

As in the 5-round attack, we consider plaintext pairs with input difference Ω_P. First, we note that if the differential is satisfied, then the input difference to each of the S-boxes 5,27, and 67 in round 6 has at most one active bit (since the difference in bits $13, 14, 79, 80, 199$, and 200 in the output of round 5 is zero by the differential). As a result, the output difference in each of these S-boxes can assume only 5 out of the 8 possible values.

Second, we observe that given a right pair, we can easily construct additional right pairs based on it. We note that bit 3 of the plaintext does not influence the bits of S-boxes 10 and 50 in round 1 and S-box 1 in round 2, which are the only S-boxes used in the differential. Hence, if (P_1, P_2) is a right pair, then $(P_1 \oplus e_3, P_2 \oplus e_3)$ is also a right pair.

Using these two observations, we can attack 6-round CTC2 in the following manner. We consider 64 plaintext pairs with difference Ω_P, and for each pair (P_1, P_2), we consider also the pair $(P_1 \oplus e_3, P_2 \oplus e_3)$. For each pair of pairs, we check whether the ciphertext differences satisfy the constraint in S-boxes 5,27, and 67. For a random pair of pairs, the probability of passing this filtering is $((5/8)^2)^3 = 2^{-4.06}$, and hence four pairs of pairs are expected to remain. These pairs are expected to contain one pair of right pairs (since the probability of the differential is $1/64$).

For each of the remaining pairs of pairs, we guess the 9 bits of the equivalent subkeys used in S-boxes 5,27, and 67 of round 6, and check whether the difference in bits $13, 14, 79, 80, 199$, and 200 in the input of round 6 is indeed zero. Each pair of pairs is expected to suggest one consistent subkey proposal on average for the nine subkey bits under attack. Thus, out of the four pairs of pairs, we expect four subkey suggestions for 9-bit information about the key. These suggestions contain

[6] In a “full” structure of 64 plaintexts, there are $64 \cdot 63/2 = 2016$ possible pairs, of which 32 satisfy the required input difference. Hence, amongst 276 pairs, we expect $32 \cdot 276/2016 = 4.38$ pairs with the required input difference.

the correct value of the subkey bits (the one suggested by the pair of right pairs). More key information can be obtained by attaching to the right pairs another "companion" pair, using other input bits that do not affect the differential (e.g., bit 4). In addition, the attacker can gain information by analyzing the S-boxes in round 6 in which the differential predicts the difference in a single input bit.

Hence, using two structures of 32 plaintexts each, it is possible to deduce the equivalent of seven subkey bits for the last round's subkey quickly and efficiently.

5.3 A Differential Attack on 7-Round CTC2

In order to mount a differential attack on a 7-round variant of the cipher, we have to extend the 5-round truncated differential presented above to 6 rounds. A natural way to extend the differential is to add one round in which the difference is "fully specified", such that the differential will consist of three constrained rounds and three unconstrained rounds. However, we checked all the possible differentials of this class and found that the number of output bits in which the difference is assured to be zero is too small, and hence cannot be used to detect the right pairs. Therefore, we use in the attack a differential consisting of four fully specified rounds and two unconstrained rounds. The first four rounds of the differential are:

$$\begin{array}{rll} \mathit{\Omega}_P = e_{30,150} & \xrightarrow{S} e_{30,151} & p = 2^{-4} \\ \xrightarrow{LT} e_{2} & \xrightarrow{S} e_{0} & p = 2^{-2} \\ \xrightarrow{LT} e_{2,123} & \xrightarrow{S} e_{0,123} & p = 2^{-4} \\ \xrightarrow{LT} e_{2,113,123,234} & \xrightarrow{S} e_{0,111,125,236} & p = 2^{-8} \end{array}$$

These four rounds are followed by two rounds where there is no restriction on the development of the difference. If the differential is satisfied, then in the output of round 6, there is zero difference in S-boxes 22 and 65, and at most one active bit in the differences of 21 more S-boxes (in total, 92 bits are assured to have zero difference).

Like the differential, the attack algorithm should also be modified. Since every input bit affects some of the S-boxes used in the differential, right pairs cannot be used to produce "companion" right pairs anymore. We compensate for that by using the stronger filtering in the last round, along with a filtering in the first round.

In the attack, we examine $2^9 = 512$ structures of plaintexts, where in each structure the input to S-boxes 10 and 50 takes all the 64 possible values, and the rest of the bits are constant. For each structure, we decrypt the ciphertexts through the linear transformation, and detect the pairs whose intermediate values satisfy the restrictions on the difference imposed by the differential (zero difference in S-boxes 22 and 65, and one of the five possible differences in 21 more S-boxes).

In each structure there are $64 \cdot 63/2 \approx 2^{11}$ pairs, 32 of which have difference e_2 after the first round. Hence, in total there are about 2^{20} pairs, 2^{14} of them may satisfy the differential in rounds 2–6. As the probability of rounds 2–6 of the differential is 2^{-14}, the obtained data is expected to contain about one right pair. The probability that a wrong pair satisfies the difference in the intermediate encryption values is

$$(2^{-3})^2 \cdot \left(\frac{5}{8}\right)^{21} = 2^{-6} \cdot 2^{-0.678 \cdot 21} = 2^{-20.2}.$$

Therefore, besides the right pair, about one wrong pair is expected to remain.

At this stage, we consider the remaining pairs and check (for each pair) whether the input difference can lead to the difference e_2 after the first round. Since only 16 of the 64 input differences satisfy this requirement, it is expected that only the right pair remains after this step. As in the 5-round attack, the right pair can be used to suggest two values for the 6 subkey bits used in S-boxes 11 and 51 in round 1. In addition, the pair suggests two values for the 3-bit subkey in each of the S-boxes in round 7, in which the input difference has exactly one active bit.[7] Hence, in total the attack reveals four bits of key information in round 1, and between 0 and 42 bits of key information in round 7.

The data complexity of the attack is 2^{15} chosen plaintexts, and the time complexity is dominated by the time required for encrypting the plaintexts (the filtering condition on the ciphertext pairs can be performed efficiently using a hash table). The memory complexity is also small, as it is sufficient to store one structure at a time.

To extend the attack to 8-round CTC2, one can extend the differential to a 7-round truncated differential, in which the differences are fully specified in the first 6 rounds, and completely unrestricted in the seventh round. This results in an attack whose data and time complexities are about 2^{60}, which is inferior to the differential-linear attack that we present in the next section.

6 Differential-Linear Attacks on CTC2

The truncated differentials and the linear approximations presented in the previous sections can be concatenated to devise differential-linear attacks on a bigger number of rounds. In this section we present differential-linear attacks on 7-round and 8-round CTC2.

6.1 A Differential-Linear Attack on 7-Round CTC2

The differential-linear attack on 7-round CTC2 is based on a 6-round distinguisher, composed of a 5-round truncated differential and a one-round linear

[7] The differential has 23 S-boxes in round 7 for which the input difference has at most one active bit. S-boxes where all the input bits have zero difference do not suggest key information, since in these S-boxes the output difference is zero, regardless of the key.

approximation. The truncated differential is the same used in the 5-round differential attack (Section 5.1), which assures that the difference in 35 bits after round 5 is zero. These 35 bits contain bits 50 and 184, and hence the differential can be combined with the following one-round linear approximation:

$$e_{50,184} \xrightarrow{S} e_{48,185} \xrightarrow{LT} e_{131} \qquad p = \frac{1}{2} - 2^{-3}$$

We note that this specific linear approximation is chosen in order to have a single active S-box in the round after the distinguisher. Since the probability of the differential is 2^{-6} and the bias of the linear approximation is 2^{-3}, the overall bias of the differential-linear approximation is $2 \cdot 2^{-6} \cdot (2^{-3})^2 = 2^{-11}$ (see [2,17]). Thus, a simple 7-round attack exploiting the distinguisher requires 2^{24} plaintext pairs with input difference Ω_P. After obtaining the ciphertexts, the attacker guesses three bits in the equivalent last round subkey, and checks whether the differential-linear approximation holds or not.

Like in the 5-round differential attack, the data complexity of the attack can be reduced using structures. The attacker considers structures of 64 plaintexts, in which the value in all the S-boxes except for S-boxes 10 and 50 is constant, and the values in these two S-boxes assume all the 64 possible values. For each guess of the six corresponding bits of the first subkey, the attacker can find 32 pairs in each structure satisfying the input difference of the second round of the differential. For these pairs, the attacker can apply a 5-round differential-linear approximation composed of the four last rounds of the differential and the same linear approximation used before. The bias of the reduced approximation is $2 \cdot 2^{-2} \cdot (2^{-3})^2 = 2^{-7}$, and thus 2^{16} pairs are sufficient for the attack. Since each structure contains 32 pairs, the attack requires 2^{11} structures, which are 2^{17} chosen plaintexts.

In order to reduce the time complexity of the attack we use an observation similar to the one presented in [3]. In the partial decryption phase of the attack, the attacker has to decrypt only a single S-box. In this S-box, there are only 64 possible pairs of ciphertext values. Hence, instead of decrypting the same values many times, the attacker can perform the following: First, the attacker counts how many times each of these 64 values occurs amongst the ciphertext pairs. Then, for each guess of the three equivalent last subkey bits, the attacker decrypts the 64 values, and checks for each of them whether the differential-linear approximation holds or not. Finally, the attacker combines the result of the check with the number of occurrences of each of the 64 values to compute the overall probability that the approximation holds, and use this probability to choose the most biased key.

Using this strategy, the time complexity of the decryption phase is negligible compared to the encryption of the plaintexts. The time complexity of the first round subkey guess is $2^6 \cdot 2 \cdot 2^{17}$ S-box computations, which are faster than 2^{17} encryptions. Hence, the overall data complexity of the attack is approximately 2^{17} encryptions.

Following [18], we can calculate the success rate of the attack to be about 81.5% (this is the probability that the right subkey is the most biased).

6.2 A Differential-Linear Attack on 8-Round CTC2

The attack on 8-round CTC2 is based on a 7-round distinguisher, composed of a 6-round truncated differential and a one-round linear approximation. The truncated differential is the same used in the 7-round differential attack (Section 5.3), which assures that the difference in 73 bits after round 6 is zero. These bits contain bit 1, and hence the differential can be combined with the following one-round linear approximation:

$$e_1 \xrightarrow{S} e_2 \xrightarrow{LT} e_{30,151} \qquad p = \frac{1}{2} - 2^{-2}$$

The probability of the differential is 2^{-18} and the bias of the linear approximation is 2^{-2}. Thus, the bias of the differential-linear approximation is $2 \cdot 2^{-18} \cdot (2^{-2})^2 = 2^{-21}$. A simple 8-round attack exploiting the 7-round differential-linear approximation requires 2^{45} chosen plaintexts.

This basic attack can be improved using structures exactly in the same manner as in the 7-round attack. This reduces the data complexity to 2^{37} chosen plaintexts, while maintaining a low time complexity dominated by encrypting the plaintexts.

Following [18], we can calculate the success rate of the attack to be about 61.8% (this is the probability that the right subkey is the most biased, with probability 69.5% the right key is amongst the two most biased keys, etc.).

7 Summary and Conclusions

In this paper we have presented several attacks on CTC2. We have showed that due to its slow diffusion, CTC2 is susceptible to standard cryptanalytic attacks. In Table 1 we summarize the various attacks, and their complexities.

Attack	Rounds	Complexity	
		Data	Time
Meet in the middle	4	4	4
Differential	5	24	24
Differential	6	64	64
Differential	7	2^{15}	2^{15}
Differential-Linear	7	2^{17}	2^{17}
Differential-Linear	8	2^{37}	2^{37}

Data complexity is given in chosen plaintexts.
Time complexity is given in encryption units.

Table 1. Summary of the Complexities of Our Attacks on CTC2

We note that our attacks rely very strongly on the very slow diffusion of the cipher. The slow diffusion, as our results exposed, may be the reason for the success of the algebraic attacks on CTC2. While this does not mean that algebraic attacks are inferior to standard attacks, it does re-open the issue of finding an example where algebraic attacks are better than standard and well understood cryptanalytic techniques.

When we compare our results to the ones in [10], we encounter some similarities. In [10], a 6-round attack with data complexity of 4 chosen plaintexts and time complexity of about 2^{253} encryptions is presented. Without discussing the methodology which was used for the time complexity estimations of [10], we note that by employing the meet-in-the-middle attack presented in Section 3.1, an attacker which is allowed only 4 chosen plaintexts can attack 6-round CTC2 with an expected speed-up of about 66%. In other words, a basic meet-in-the-middle attack, which probably can be further optimized (e.g., there are key considerations which may be useful for improving the attack) achieves a 66% speed-up, to be compared with the 75% speed-up gained in [10].

If only meet-in-the-middle attacks given 4 chosen plaintexts are taken into consideration, it seems that 4-round CTC2 can be easily broken, while 6-round CTC2 requires an exponential time (equivalent to about 2^{253} trial encryptions). Even though time estimates for algebraic attacks on 4-round CTC2 are not given, it would be surprising if such attacks require time complexity which is of the order of magnitude of exhaustive key search. Hence, it seems that the strength of the algebraic technique in attacking 4-round or 6-round CTC2 is approximately equivalent to the strength of the meet-in-the-middle technique.

This leaves the issue of attacking 5-round CTC2 as a challenge for the algebraic technique. We believe that given 4 chosen plaintexts, the best attack on 5-round CTC2 using a simple meet-in-the-middle technique, is the variant of exhaustive search presented in Section 3.1. If algebraic techniques can beat these results by an order of magnitude, this would look as a proof of their merits (assuming of course, that no one finds a better attack using easy to analyze attack methods).

References

1. Eli Biham, *On Matsui's Linear Cryptanalysis*, Advances in Cryptology, Proceedings of Eurocrypt 1994, Lecture Notes in Computer Science 950, pp. 341–355, Springer, 1994.
2. Eli Biham, Orr Dunkelman, Nathan Keller, *Enhancing Differential-Linear Cryptanalysis*, Advances in Cryptology, proceeding of ASIACRYPT 2002, Lecture Notes in Computer Science 2501, pp. 254–266, Springer, 2002.
3. Eli Biham, Orr Dunkelman, Nathan Keller, *Differential-Linear Cryptanalysis of Serpent*, proceedings of Fast Software Encryption 10, Lecture Notes in Computer Science 2887, pp. 9–21, Springer-Verlag, 2003.
4. Alex Biryukov, Christophe De Cannière, Michaël Quisquater, *On Multiple Linear Approximations*, Advances in Cryptology, Proceedings of CRYPTO 2004, Lecture Notes in Computer Science 3152, pp. 1–22, Springer, 2004.
5. Johannes Buchmann, Andrei Pyshkin, Ralf-Philipp Weinmann, *Block Ciphers Sensitive to Groebner Basis Attacks*, Proceedings of CT-RSA 2006, Lecture Notes in Computer Science 3860, pp. 313–331, Springer, 2006.

6. Baudoin Collard, François-Xavier Standaert, Jean-Jacques Quisquater, *Improved and Multiple Linear Cryptanalysis of Reduced Round Serpent*, Proceedings of Inscrypt 2007, Lecture Notes in Computer Science 4990, pp. 51–65, Springer, 2008.
7. Nicolas T. Courtois, *Fast Algebraic Attacks on Stream Ciphers with Linear Feedback*, Advances in Cryptology, Proceedings of CRYPTO 2003, Lecture Notes in Computer Science 2729, pp. 176–194, Springer, 2003.
8. Nicolas T. Courtois, *How Fast can be Algebraic Attacks on Block Ciphers?*, IACR ePrint report 2006/168, 2006.
9. Nicolas T. Courtois, *Algebraic Attacks On Block Ciphers*, Dagstuhl seminar on Symmetric Cryptography, January 2007.
10. Nicolas T. Courtois, *CTC2 and Fast Algebraic Attacks on Block Ciphers Revisited*, IACR ePrint report 2007/152, 2007.
11. Nicolas T. Courtois, Gregory V. Bard, David Wagner, *Algebraic and Slide Attacks on KeeLoq*, Proceedings of Fast Software Encryption 15, Lecture Notes in Computer Science 5086, pp. 97–115, Springer, 2008.
12. Nicolas T. Courtois, Willi Meier, *Algebraic Attacks on Stream Ciphers with Linear Feedback*, Advances in Cryptology, Proceedings of EUROCRYPT 2003, Lecture Notes in Computer Science 2656, pp. 345–359, Springer, 2003.
13. Nicolas T. Courtois, Josef Pieprzyk, *Cryptanalysis of Block Ciphers with Overdefined Systems of Equations*, Advances in Cryptology, Proceedings of ASIACRYPT 2002, Lecture Notes in Computer Science 2501, pp. 267–287, Springer, 2002.
14. Orr Dunkelman, Nathan Keller, *Linear Cryptanalysis of CTC*, IACR ePrint report 2006/250, 2006.
15. Orr Dunkelman, Gautham Sekar, Bart Preneel, *Improved Meet-in-the-Middle Attacks on Reduced-Round DES*, Proceedings of Indocrypt 2007, Lecture Notes in Computer Science 4859, pp. 86–100, Springer, 2007.
16. Jean-Charles Faugére, Ludovic Perret, *Algebraic Cryptanalysis of Curry and Flurry using Correlated Messages*, IACR ePrint report 2008/402, 2008.
17. Susan K. Langford, *Differential-Linear Cryptanalysis and Threshold Signatures*, Ph.D. thesis, 1995.
18. Ali Aydin Selçuk, *On Probability of Success in Linear and Differential Cryptanalysis*, Journal of Cryptology, vol. 21, no. 1, pp. 131–147, Springer, 2008.