New Treatment of the BSW Sampling and Its Applications to Stream Ciphers*

Lin Ding[1], Chenhui Jin[1], Jie Guan[1], and Chuanda Qi[2]

[1]Information Science and Technology Institute, 450000 Zhengzhou, China
[2]Xinyang Normal University, 464000 Xinyang, China
{dinglin_cipher@163.com,jinchenhui@126.com,guanjie007@163.com,
qichuanda@sina.com}

Abstract. By combining the time-memory-data tradeoff (TMDTO) attack independently proposed by Babbage and Golić (BG) with the BSW sampling technique, this paper explores to mount a new TMDTO attack on stream ciphers. The new attack gives a wider variety of trade-offs, compared with original BG-TMDTO attack. It is efficient when multiple data is allowed for the attacker from the same key with different IVs, even though the internal state size is twice the key size. We apply the new attack to MICKEY and Grain stream ciphers, and improves the existing TMDTO attacks on them. Our attacks on Grain v1 and Grain-128 stream ciphers are rather attractive in the respect that the online time, offline time and memory complexities are all better than an exhaustive key search, and the amount of keystream needed are completely valid. Finally, we generalize the new attack to a Guess and Determine-TMDTO attack on stream ciphers, and mount a Guess and Determine-TMDTO attack on SOSEMANUK stream cipher with the online time and offline time complexities both equal to 2^{128}, which achieves the best time complexity level compared with all existing attacks on SOSEMANUK so far.

Keywords: Cryptanalysis; Time-memory-data tradeoff attack; BSW sampling; Guess and Determine attack; Stream cipher; MICKEY; Grain; SOSEMANUK.

1 Introduction

Stream ciphers can be described as keyed generators of pseudo random sequences over a finite field. Usually, the problem of recovering the secret key of stream cipher can be generalized as the problem of inverting a one-way function $y = f(x)$. Exhaustive key search and table lookup attack are two extreme examples of generic attacks to invert one-way functions. Time-Memory tradeoff (TMTO) attack is a method combining the exhaustive key search and the table lookup

* This work is supported in part by the National Natural Science Foundation of China (No. 61202491, 61272041, 61272488) and Foundation of Science and Technology on Information Assurance Laboratory (Grant No. KJ-13-007). This is the full version of the paper to appear in the Proceedings of AFRICACRYPT 2014.

attack, and offers a generic technique to invert one-way functions, where one can trade off time and memory costs. A typical TMTO attack consists of two phases, i.e., the offline (or pre-computation) phase and the online phase. The complexities of TMTO attack can be evaluated by looking at three main parameters, i.e., the online time complexity T, the memory cost M, and the offline time complexity P.

The idea of TMTO attack was originally proposed by Hellman [1] for attacking the DES block cipher. The attack has a lower time complexity (in online phase) than the exhaustive key search and a lower memory complexity than the table lookup attack. Its tradeoff curve is obtained as $TM^2 = N^2$ and $P = N$, where N is the number of possible keys. Hence, a reasonable choice of M and T is $T = M = N^{2/3}$, which is lower than the exhaustive key search. However, the offline time complexity of Hellmans attack is always no less than the time complexity of the exhaustive key search.

Babbage [2] and Golić [3] independently proposed a simple time-memory-data tradeoff (TMDTO) attack on stream ciphers. The tradeoff curve of Babbage-Golić (BG-TMDTO) attack can be represented as $TM = N$, $P = M$ and $T = D$, where D is the amount of data available to the attacker. Here, N is the number of possible internal states. In ASIACRYPT 2000, Biryukov and Shamir [4] found that TMTO attacks against stream ciphers can be extended to TMDTO attacks by utilizing multiple data points. The idea of Biryukov-Shamir (BS-TMDTO) attack was similar to the original attack by Hellman. The tradeoff curve of this attack can be represented as $TM^2D^2 = N^2$ and $P = N/D$, while the restriction $1 \leq D^2 \leq T$ has to be satisfied. The attack was applied to one of the most widely deployed stream ciphers, GSM's A5/1. In these attacks, the attacker tries to invert the function mapping the internal state of stream cipher to a segment of the keystream output. These attacks imply that the state size should be at least twice the key size, which is widely considered as an essential design principle for modern stream ciphers.

On a different attack scenario, TMDTO attacks can also be used to invert the function mapping the initial inputs (e.g., Key and IV) of stream cipher to a segment of the keystream output. In the TMDTO attack proposed by Hong and Sarkar [5], they treat both the secret key and IV as unknown in the offline phase. The trade-off curve for Hong-Sarkar (HS-TMDTO) attack is the same as the BS-TMDTO attack with $N = K \times V$. Here, denote K and V the numbers of possible keys and IVs respectively.

In [6], Dunkelman and Keller presented a new approach to TMDTO attacks against stream ciphers. They did not treat the IV as part of the secret key material, after exploiting the fact that the IV is known during an online attack. The Dunkelman-Keller (DK-TMDTO) attack get the same trade-off curve as $TM^2D^2 = N^2$ and $P = N/D$, but with the restriction $1 \leq D^2 \leq T$ replaced by the restrictions $V \geq D$ and $T \geq D$. The attack implies that if the key length is n bits, the cipher offers n-bit security with respect to DK-TMDTO attacks only if the IV length is at least $1.5n$ bits. However, its offline time complexity cannot be faster than exhaustive key search irrespective of the IV length.

The concept of BSW Sampling was introduced by Biryukov, Shamir and Wagner [7] at FSE 2000. It helps the BS-TMDTO attack to get a wider choice of parameters by relaxing its restriction. However, a view exists that the BG-TMDTO attack is not really helped as much by the BSW sampling technique [8]. This paper explores the possibility of combining the BG-TMDTO attack with the BSW sampling technique to mount a new TMDTO attack on stream ciphers. The new attack can be considered as a generalization of BG-TMDTO attack. It is rather efficient when multiple data is allowed for the attacker from the same key with different IVs, even though the internal state size is twice the key size. As applications, we mount new TMDTO attacks on MICKEY and Grain stream ciphers. Finally, a general Guess and Determine-TMDTO attack on stream ciphers is presented by exploiting the fact that the BSW sample technique can be generalized to a simple Guess and Determine attack. As an application, we give a Guess and Determine-TMDTO attack on SOSEMANUK stream cipher, which achieves the best time complexity level compared with all existing attacks on SOSEMANUK so far.

This paper is organized as follows. Our new attack is proposed in Section 2. Applications of the new attack and some discussions on it are given in Section 3. The paper is concluded in Section 4.

2 New Time-Memory-Data Tradeoff Attack

2.1 BSW Sampling Technique

Inverting a one-way function has an important role in the security of most encryption schemes. The problem of recovering the internal state of a stream cipher, given multiple keystream segments, can be generalized as an inversion problem, as shown as follows.

Given a one-way function $f : X \to Y$ and a set $D = \{y_i\}_i \subset Y$, to identify an internal state $x \in X$ such that $f(x) = y_i$ for some *i*.

Where $N = |X|$ denotes the number of possible internal states, and Y the set of enough keystream segments.

The BSW sampling aims at obtaining wider choices of tradeoff parameters for the tradeoff curve of BS-TMDTO attack. Its main idea is to find an efficient way to generate and enumerate special cipher states, from which the first subsequent keystream output bits of the cipher are a fixed string (such as a run of consecutive 1 or 0 bits). If this can be done for a run of l bits, the sampling resistance of the cipher is defined to be $R = 2^{-l}$. Usually, the BSW sampling tradeoff works when the following assumption is satisfied for a given stream cipher.

Assumption 1. For a given stream cipher with the internal state size $n = \log_2 N$, given the value of $n-l$ particular state bits of it and the first l keystream bits produced from that state, the remaining l internal state bits may be deduced directly.

It is easy to see that the sampling resistance of the given stream cipher is $R = 2^{-l}$ under the assumption. Given the sampling resistance, the attacker can apply the BS tradeoff to the problem of inverting the restricted function $f' : X' \to Y'$, rather than the function $f : X \to Y$. The restricted function is obtained as follows.

---

1. Fix a specific function by choosing an l-bit string S.
2. Given an $(n-l)$-bit input value x, treat S as the first l bits of keystream, compute the remaining l state bits according to the assumption above, and then expand it to n bits.
3. Clock the stream cipher n steps, generating an n-bit keystream segment $S|y$.
4. Output y.

Clearly, the inversion problem of inverting the function f mapping n-bit states to n-bit keystream segments is equivalent to the inversion problem of inverting the restricted function $f' : \{0,1\}^{n-l} \to \{0,1\}^{n-l}$. Thus, the attacker would consider the cost of inverting f' rather than the full stream cipher.

The trade-off curve for BSW sampling is the same as the BS-TMDTO attack, i.e., $TM^2D^2 = N^2$ and $P = N/D$, while a wider choice of parameters by relaxing the restriction $1 \leq D^2 \leq T$ to $1 \leq R^2D^2 \leq T$. The BSW sampling technique has been applied to MICKEY and Grain stream ciphers, see [9, 8] for more details. The BSW sampling technique allows mounting the BS-TMDTO attack in a larger variety of settings by relaxing the restriction, though the obtained tradeoff curve keeps unchanged.

2.2 New Treatment of the BSW Sampling Technique

As shown above, the BSW sampling technique helps the BS-TMDTO attack to get a wider choice of parameters by relaxing its restriction. However, claimed by [8], the BG tradeoff is not really helped as much by the BSW sampling, as the amount of keystream needed remains prohibitive. In fact, if the keystream segments available for cryptanalysis are generated by only one single (K, IV), the required amount of keystream is indeed prohibitive. However, if the proposed cryptanalysis does not require a long keystream generated by one single (K, IV) pair but a (large) number of short keystreams generated by the same key with different IVs, the required amount of keystream may be completely valid in this respect. This motivates us to explore the possibility of combining the BG-TMDTO attack with the BSW sampling technique to mount a new TMDTO attack. In this attack, the keystream segments available for cryptanalysis are generated by the same key with different IVs.

Assume that an attacker can collect a set of d keystream sequences generated by the given stream cipher for different IVs, and that the length of each sequence is d'. Accordingly, we assume that the set of samples available for the cryptanalysis consists of approximately $D = d \cdot d'$ $2n$-bit keystream segment samples.

Like the typical TMDTO attack, the new attack consists of two phases, i.e., the offline phase and the online phase. The attack utilizes an integer parameter r satisfying the restriction $1 \leq r \leq R^{-1}$.

The offline phase is to construct some tables consisting of pairs of internal state and corresponding keystream segment. The algorithm for the offline phase is described as follows.

The Offline Algorithm

Choose r strings $S_1, \cdots, S_r$ randomly, and each consists of l bits. For each fixed string S_i, do the followings.

1. Choose N' strings $I_1, \cdots, I_{N'}$ randomly, and each consists of $n - l$ bits.
2. Treat S_i as the first l bits of keystream and I_j as the $n-l$ particular state bits, compute the remaining l bits, clock the stream cipher n steps to generate an n-bit keystream segment, and then memory the (n-bit internal state, n-bit keystream segment) pair in the table T_i.

The online phase is to recover an internal state which has generated a keystream segment in one table. The algorithm for the online phase is described as follows.

The Online Algorithm

For each $2n$-bit keystream segment sample available for the cryptanalysis, check if the first l bits of the sample match one of r strings $S_1, \cdots, S_r$. If a matching is not found, go to check the next $2n$-bit keystream segment sample. If a matching is found, do the followings.

1. For each matching (say S_i) found, check if the first n-bit keystream segment exists in the second column of the corresponding table (i.e., T_i). If it does not exist, move to consideration of the next $2n$-bit keystream segment sample. If it exists, read the corresponding n-bit internal state in the first column of the table, clock the stream cipher $2n$ steps to generate $2n$ keystream bits, and then match them with the sample. If the match passes, go to the output (a). Otherwise, move to consideration of the next $2n$-bit keystream segment sample.
2. If no more keystream segment samples, go to the output (b).

Output: (a) recovered n-bit internal state; (b) a flag that the algorithm has failed.

The complexities of the proposed TMDTO attack are calculated as follows.

In the offline phase, for each fixed string S_i, N' strings should be chosen randomly to execute the Step 2. Thus, the time complexity of the offline phase

(denoted as P) is mainly determined by the number of (S_i, I_j), which implies that $P = rN'$. At the same time, the memory complexity of the offline phase (denoted as M) is the same with the offline time complexity, i.e., $M = P = rN'$, since the offline phase has to construct r tables, and that the size of each table is N'.

Clearly, a total of rN' (n-bit internal state, n-bit keystream segment) pairs has been stored in the offline phase, since N' strings are chosen randomly for each of all r strings $S_1, \cdots, S_r$ in this phase. According to the birthday paradox, the expected number of keystream segment samples available for the online phase should be $D = N/rN'$. Since the probability that the first l bits of a given sample match one of r strings $S_1, \cdots, S_r$ is $p = r \cdot 2^{-l} = rR$, the expected number of matchings found among all D keystream segment samples should be $D \cdot p = RN/N'$. It is easy to see that the time complexity of the online phase (denoted as T) is mainly determined by the number of matchings found among all samples, which implies that $T = D \cdot p = RN/N'$. Note that the unit of the online time complexity is one table lookup. Therefore, the trade-off curve of our attack is given as follows.

$$MT = rRN,\ MD = N,\ P = M \text{ and } D = d \cdot d'$$

Where r is integer parameter satisfying the restriction $1 \leq r \leq R^{-1}$.

Clearly, the tradeoff curve of our attack is the same as the BG-TMDTO attack when $r = R^{-1}$ holds. Thus, the BG-TMDTO attack may be considered as a special case of our attack. By introducing the integer parameter, our attack gives a wider variety of trade-offs. Let $D_{\max} = d_{\max} \cdot d'_{\max}$ be the maximum number of keystream bits generated by the key K, where $d'_{\max}$ denotes the maximum number of keystream bits generated by a single (K, IV), and $d_{\max}$ denotes the maximum number of keystream sequences generated by the same key K and different IVs. A lemma is obtained as follows.

Lemma 1. For a given stream cipher with the internal state size of $\log_2 N$ bits, when $N^{1/2} < D_{\max} < (rR)^{-1}N^{1/2}$ is allowed for the attacker, $1 \leq r < R^{-1}$, there certainly exists a TMDTO attack with the online time, offline time and memory complexities all faster than an exhaustive key search on the cipher, even though the internal state size is twice the key size.

Proof. Recall the trade-off curve of our attack as follows.

$$MT = rRN,\ MD = N,\ P = M \text{ and } D = d \cdot d'$$

When $N^{1/2} < D_{\max} < (rR)^{-1}N^{1/2}$ is allowed for the attacker, we may choose the data complexity such that $N^{1/2} < D \leq D_{\max}$, which implies

$$rRN^{1/2} < P = M = \tfrac{N}{D} < N^{1/2}$$
$$T = \tfrac{rRN}{M} < \tfrac{rRN}{rRN^{1/2}} = N^{1/2}$$

Thus, this Lemma follows directly. ■

Our attack can be considered as a generalization of the BG-TMDTO attack. It is rather efficient when multiple data is allowed for the attacker from the same key with different IVs, even though the internal state size is twice the key size.

3 Applications and Discussions

3.1 Previous Works on MICKEY and Grain Stream Ciphers

At FSE 2000, particularly efficient attacks on A5/1 were proposed by using the BSW sampling technique [7]. After then, the BS-TMDTO attack with BSW sampling had been applied to MICKEY 1.0 and Grain stream ciphers, see [9, 8].

MICKEY 1.0 [10] is a hardware-oriented stream cipher proposed by Babbage and Dodd in 2005. Its strengthened version, named MICKEY 2.0 [11], had been selected as one of the seven finalists of eSTREAM project. In [9], Hong and Kim showed that MICKEY 1.0 stream cipher has a sampling resistance of at most 2^{-27}. Since MICKEY 1.0 has an internal state size of 160 bits, the BS-TMDTO attack with BSW sampling on the cipher has the online and offline time complexities of 2^{67} and 2^{100} respectively, while the time complexity of an exhaustive key search is 2^{80}. The attack is not applicable to MICKEY 2.0, because the internal state size was increased from 160 to 200 bits. MICKEY-128 2.0 [11] is a variant of MICKEY 2.0, which supports a key size of 128 bits, and an IV varying between 0 and 80 bits in length. Its internal state size is of 320 bits.

Grain v1 [12], a hardware-oriented stream cipher proposed by Hell, Johansson and Meier, is also one of the seven eSTREAM finalists. It has a key size of 80 bits, an IV size of 64 bits, and an internal state size of 160 bits. In [8], Bjørstad showed that Grain v1 stream cipher has a sampling resistance of at most 2^{-21}. The BS-TMDTO attack with BSW sampling on the cipher has the online and offline time complexities of 2^{70} and 2^{104} respectively. Grain-128 [12] is a variant of Grain, which supports a key size of 128 bits, and an IV size of 96 bits. Its internal state size is of 256 bits.

3.2 New Attacks on MICKEY and Grain Stream Ciphers

Now, we will apply our attack to MICKEY and Grain stream ciphers. MICKEY 1.0 has a key size of 80 bits, and supports an IV varying between 0 and 80 bits in length. As for MICKEY 1.0 stream cipher, we get that $N = 2^{160}$ and $R = 2^{-27}$. In the specification of MICKEY 1.0, there exists a restriction that the maximum length of keystream sequence that may be generated with a single (K, IV) pair is 2^{40} bits, and it is acceptable to generate 2^{40} such sequences, all from the same key K but with different values of IV. This restriction implies $d \leq 2^{40}$ and $d' \leq 2^{40}$. Here, we choose $d = d' = 2^{40}$, and $r = 1$. Since $N = 2^{160}$

and $R = 2^{-27}$ are known, we can mount a TMDTO attack on MICKEY 1.0 with $M = D = P = N^{1/2} = 2^{80}$ and $T = 2^{53}$.

Similarly, we can mount TMDTO attacks on MICKEY 2.0, MICKEY-128 2.0, Grain v1 and Grain-128 stream ciphers. The results and comparisons with the existing attacks are summarized in Table 1.

Table 1. The results and comparisons with the existing attacks

Stream ciphers	R	Attacks	Parameter	T	M	D	P
MICKEY 1.0	2^{-27}	[9]	-	2^{67}	2^{67}	2^{60}	2^{100}
		This paper	$r = 1$	2^{53}	2^{80}	$d = d' = 2^{40}$	2^{80}
MICKEY 2.0	2^{-33}	This paper	$r = 1$	2^{47}	2^{120}	$d = d' = 2^{40}$	2^{120}
MICKEY-128 2.0	2^{-54}	This paper	$r = 1$	2^{74}	2^{192}	$d = d' = 2^{64}$	2^{192}
Grain v1	2^{-21}	[8]	-	2^{70}	2^{59}	2^{56}	2^{104}
		This paper	$r = 1$	$2^{69.5}$	$2^{69.5}$	$d = d' = 2^{45.25}$	$2^{69.5}$
		This paper	$r = 2^{11}$	2^{75}	2^{75}	$d = d' = 2^{42.5}$	2^{75}
Grain-128	2^{-22}	This paper	$r = 1$	2^{117}	2^{117}	$d = d' = 2^{69.5}$	2^{117}
		This paper	$r = 2^{12}$	2^{123}	2^{123}	$d = d' = 2^{66.5}$	2^{123}

Note that similar to MICKEY 2.0, there also exists a restriction for MICKEY-128 2.0 that the maximum length of keystream sequence that may be generated with a single (K, IV) pair is 2^{64} bits, and it is acceptable to generate 2^{64} such sequences, all from the same key K but with different values of IV. Thus, the restrictions $d \leq 2^{64}$ and $d' \leq 2^{64}$ should hold simultaneously. While, as for Grain v1 and Grain-128 stream ciphers, their specifications do not specify any limits to the amount of keystream that may be generated by the same key K with different IVs, so our attacks on them are completely valid in this respect.

According to Table 1, our TMDTO attacks have one remarkable advantage in the online time complexity, which is always better than an exhaustive key search. Since the offline phase only performs once, the attack with a low online time complexity works, particularly when the attacker wants to recover many internal states generated by different secret keys in the online phase. Furthermore, as for Grain v1 and Grain-128 stream ciphers, our attacks are rather attractive in the respect that the online time, offline time and memory complexities are all better than an exhaustive key search, and the amount of keystream needed are completely valid.

3.3 General Guess and Determine-TMDTO Attack

The Guess and Determine (GD) attack is a common attack on stream ciphers. Its main idea is to guess a portion of the internal state, and then to recover the remaining internal state by using a small amount of known keystream. Clearly, the BSW sample technique can be generalized to a simple Guess and Determine attack. The Assumption 1 can be rewritten as follows.

Assumption 2. For a given stream cipher with internal state size $n = \log_2 N$, the attacker guesses the values of $n-l$ particular internal state bits, the remaining l internal state bits may be determined directly by using the first l keystream bits produced from that state.

Assume that for a given stream cipher, the attacker obtains a simple Guess and Determine attack, described as the Assumption 2. The attack has a time complexity of 2^{n-l}, requiring l keystream bits. It can be transformed into a Guess and Determine-TMDTO attack by fitting it into the model showed in Subsection 2.2.

Take the SOSEMANUK stream cipher [13], one of seven eSTREAM finalists, for example. SOSEMANUK has an internal state size of 384 bits. The internal state of SOSEMANUK at time t can be showed as $s_{t+1}, \cdots, s_{t+10}, R1_{t+1}, R2_{t+1}$, and that each contains 32 bits. Its key length is variable between 128 and 256 bits. It accommodates a 128-bit IV. Any key length is claimed to achieve 128-bit security. The best Guess and Determine attack on the cipher so far has been proposed by Feng et al. [14] in ASIACRYPT 2010, with a time complexity of 2^{176}. In their attack, the attacker guesses a total of 176 internal state bits, and then determine the remaining 208 internal state bits by using eight 32-bit keystream words (i.e., a total of 256 keystream bits). The attack consists of five phases. Their attack can not be transformed into a Guess and Determine-TMDTO attack, since it requires too many keystream bits that it does not fit into our model. Here, we give a new Guess and Determine attack on SOSEMANUK as follows. We follow the notes used in [14]. Let x be a 32-bit word. Denote $x^{(i)}$ the i-th byte component of x, $0 \leq i \leq 3$, i.e., $x = x^{(3)}||x^{(2)}||x^{(1)}||x^{(0)}$, where each $x^{(i)}$ is a byte, and $||$ is the concatenation of two bit strings. For simplicity we write $x^{(1)}||x^{(0)}$ as $x^{(0,1)}$ and $x^{(2)}||x^{(1)}||x^{(0)}$ as $x^{(0,1,2)}$.

In the new attack, we should first guess $s_1, s_2, s_3, s_4^{(0)}, R2_1^{(0,1,2)}$ and $R1_1$ (a total of 160 bits). Note that in their attack, they make an assumption on the least significant bit of $R1_1$. As showed in the Section 6 of [14], it shows that the assumption is not necessary for their attack to work. For convenience, we guess the least significant bit of $R1_1$ directly instead of making one assumption. After then, we determine $s_4, s_5, s_6, R2_1$ and s_{10} by executing the Phase 1-3 of their attack. Finally, the attacker should guess s_7, s_8 and s_9 (a total of 96 bits). Up to now, we have recovered all 384 internal state bits of SOSEMANUK. The attack only uses four 32-bit keystream words, i.e., z_1, z_2, z_3 and z_4.

In the new attack, we should guess $s_1, s_2, s_3, s_4^{(0)}, R2_1^{(0,1,2)}, s_7, s_8, s_9$ and $R1_1$ (a total of 256 bits), and then recover the remaining 128 internal state bits by using four 32-bit keystream words (a total of 128 bits). The new attack fits into our model quite well. Thus, we have $N = 2^{384}$ and $R = 2^{-128}$ for SOSEMANUK stream cipher. Our results and comparison with the existing attacks are summarized in Table 2.

Note that the specification of SOSEMANUK does not specify any limits to the amount of keystream that may be generated under a single (K, IV), so our attacks are completely valid in this respect. According to Table 2, the attacker

Table 2. The results and comparisons with the existing attacks on SOSEMANUK

Attacks	Parameter	T	M	D	P
GD attack [14]	-	2^{176}	-	2^4	-
Linear Cryptanalysis [15]	-	$2^{147.9}$	$2^{147.1}$	$2^{145.5}$	-
Linear Cryptanalysis [16]	-	$2^{147.4}$	$2^{146.8}$	$2^{135.7}$	-
This paper	$r = 1$	2^{136}	2^{120}	$d = 2^{128}, d' = 2^{136}$	2^{120}
This paper	$r = 1$	2^{128}	2^{128}	$d = d' = 2^{128}$	2^{128}
This paper	$r = 1$	2^{116}	2^{140}	$d = d' = 2^{122}$	2^{140}

can mount a Guess and Determine-TMDTO attack with the online time, offline time and memory complexities all equal to 2^{128}, which achieves the best time complexity level compared with all existing attacks on SOSEMANUK so far. Of course, one can also mount a Guess and Determine-TMDTO attack with an online time complexity of 2^{116}, which is significantly better than an exhaustive key search, at the cost of increased offline time and memory complexities.

4 Conclusions

By combining the BG-TMDTO attack with the BSW sampling technique, this paper proposes a new TMDTO attack on stream ciphers. The results show that the new attack gives a wider variety of trade-offs, compared with original BG-TMDTO attack, and is efficient when multiple data is allowed for the attacker from the same key with different IVs, even though the internal state size is twice the key size. As applications, we mount TMDTO attacks on MICKEY and Grain stream ciphers. Particularly, as for Grain v1 and Grain-128 stream ciphers, our TMDTO attacks are rather attractive in the respect that the online time, offline time and memory complexities are all better than an exhaustive key search, and the amount of keystream needed are completely valid. The results are sufficient evidences of validity of our attack. Finally, we generalize our attack to a Guess and Determine-TMDTO attack, and apply it to SOSEMANUK stream cipher, which achieves the best time complexity level compared with all existing attacks on SOSEMANUK so far. We hope that our attack provides some new insights on TMDTO attacks on stream ciphers.

Acknowledgements. The authors would like to thank the anonymous reviewers and Dr. Long Wen for their valuable comments and suggestions.

References

1. Hellman, M.: A cryptanalytic time-memory trade-off. IEEE Transactions on Information Theory. 26(4), 401-406 (1980)
2. Babbage, S.: Improved exhaustive search attacks on stream ciphers. In: European Convention on Security and Detection 1995, IEE Conference Publication, pp. 161-166. IEEE Press, New York (1995)

3. J. Golić. Cryptanalysis of alleged A5 stream cipher. In: Fumy, W. (eds.) EUROCRYPT 1997. LNCS, vol. 1233, pp. 239–255. Springer, Heidelberg (1997)
4. Biryukov, A., Shamir, A.: Cryptanalytic time/memory/data tradeoffs for stream ciphers. In: Okamoto, T. (eds.) ASIACRYPT 2000. LNCS, vol. 1976, pp. 1-13. Springer, Heidelberg (2000)
5. Hong, J., Sarkar, P.: New Applications of Time Memory Data Tradeoffs. In: Roy, B. (eds.) ASIACRYPT 2005. LNCS, vol. 3788, pp. 353–372. Springer, Heidelberg (2005)
6. Dunkelman, O., Keller, N.: Treatment of the initial value in Time-Memory-Data Trade-off attacks on stream ciphers. Information Processing Letters. 107(5), pp. 133–137. (2008)
7. Biryukov, A., Shamir, A., Wagner, D.: Real time cryptanalysis of A5/1 on a PC. In: Schneier, B. (eds.) FSE 2000. LNCS, vol. 1978, pp. 1–18. Springer, Heidelberg (2001)
8. Bjørstad, T.E.: Cryptanalysis of Grain using Time/Memory/Data Tradeoffs. ECRYPT Stream Cipher Project Report 2008/012 (2008), http://www.ecrypt.eu.org/stream
9. Hong, J., Kim, W.H.: TMD-Tradeoff and State Entropy Loss Considerations of Streamcipher MICKEY. In: Maitra, S., Madhavan, C., Venkatesan, R. (eds.) INDOCRYPT 2005. LNCS, vol. 3797, pp. 169–182. Springer, Heidelberg (2005)
10. Babbage, S., Dodd, M.: The stream cipher MICKEY (version 1). ECRYPT Stream Cipher Project Report 2005/015, 2005, http://www.ecrypt.eu.org/stream
11. Babbage, S., Dodd, M.: The MICKEY Stream Ciphers. In: Robshaw, M., Billet, O. (eds.) New Stream Cipher Designs. LNCS, vol. 4986, pp. 191–209. Springer, Heidelberg (2008)
12. Hell, M., Johansson, T., Maximov, A., Meier, W.: The Grain Family of Stream Ciphers. In: Robshaw, M., Billet, O. (eds.) New Stream Cipher Designs. LNCS, vol. 4986, pp. 179–190. Springer, Heidelberg (2008)
13. Berbain, C., Billet, O., Canteaut, A., Courtois, N.T., Gilbert, H., Goubin, L., Gouget, A., Granboulan, L., Lauradoux, C., Minier, M., Pornin, T., Sibert, H.: Sosemanuk, a Fast Software-Oriented Stream Cipher. In: Robshaw, M., Billet, O. (eds.) New Stream Cipher Designs. LNCS, vol. 4986, pp. 98–118. Springer, Heidelberg (2008)
14. Feng, X., Liu, J., Zhou, Z., Wu, C., Feng, D.: A Byte-Based Guess and Determine Attack on SOSEMANUK. In: Abe, M. (eds.) ASIACRYPT 2010. LNCS, vol. 6477, pp. 146–157. Springer, Heidelberg (2010)
15. Lee, J., Lee, D., Park, S.: Cryptanalysis of SOSEMANUK and SNOW 2.0 using linear masks. In: Pieprzyk, J. (eds.) ASIACRYPT 2008. LNCS, vol. 5350, pp. 524–538. Springer, Heidelberg (2008)
16. Cho, J.Y., Hermelin, M.: Improved Linear Cryptanalysis of SOSEMANUK. In: Lee, D., Hong, S. (eds.)ICISC 2009. LNCS, vol. 5984, pp. 101–116. Springer, Heidelberg (2010)