Cryptanalysis of Grain using Time / Memory / Data Tradeoffs

T.E. Bjørstad

The Selmer Center, Department of Informatics,
University of Bergen, Pb. 7800, N-5020 Bergen, Norway.
Email : tor.bjorstad@ii.uib.no

Abstract. Grain is a hardware-oriented stream cipher designed by Hell et al., which has been selected as one of three hardware portfolio ciphers by eSTREAM, the ECRYPT Stream Cipher Project. Time / memory / data tradeoffs are a class of generic attacks used to invert general one-way functions. We show that Grain has a low resistance to so-called BSW-sampling, leading to generic tradeoffs that in the active phase recover the internal state of Grain v1 with parameters such as $O(2^{70})$ time, $O(2^{69})$ memory, and $O(2^{56})$ bits of known keystream. While the practical significance of these attacks may be arguable, these attacks violate design assumptions made in the Grain v1 specification. A similar attack applies to Grain-128.

1 Introduction

The eSTREAM [8] project was a multi-year effort to develop a portfolio of promising new stream cipher designs. Sponsored by the ECRYPT Network of Excellence, the project started in the fall of 2004 with a call for primitives, which resulted in 34 submitted designs. The stream ciphers were intended to satisfy either a software-oriented or a hardware-oriented profile, and provide some advantage over AES in at least one significant aspect. After three evaluation rounds, eSTREAM ended in the spring of 2008 with a final portfolio consisting of 8 ciphers [8]. The portfolio has since been reduced to 7 stream ciphers, due to the successful cryptanalysis of F-FCSR [11].

Grain [13] was a submission in the hardware category by Hell, Johansson and Meier. After attacks were found against the initial design, a tweaked version called Grain v1 [12] was submitted by the authors. The cipher attracted a lot of attention due to its compact design and parallelizability, enabling implementors to make different tradeoffs between speed, gate count and power consumption [10]. There is also a 128-bit version of Grain called Grain-128 [14]. In the context of this article, "Grain" refers to the Grain v1 specification [12].

1.1 Time / Memory / Data Tradeoffs

Time / memory (TM) tradeoffs were first introduced by Hellman [15] in 1980 as a generic way of attacking block ciphers, but can be generalised to the general problem of inverting one-way functions. In the case of function invertion (but not for block ciphers), Babbage and Golic [1, 9] and later Biryukov, Shamir and Wagner [4, 5] showed that the basic TM tradeoffs can be improved significantly by utilising several data points. This class of tradeoffs are thus known as time / memory / data (TMD) tradeoffs, and use the birthday paradox to ensure that a preimage will be found for at least one of the data points with high probability. For stream ciphers, the one-way function to be inverted is commonly taken to be the map sending an n-bit internal state of the cipher to the first n bits of keystream generated from that state. Several stream ciphers have been broken by TMD tradeoffs, most famously the GSM encryption scheme A5/1 [5].

Using a TMD tradeoff to invert a function can be split into two phases; a preprocessing step followed by an active phase. In the preprocessing step, which is performed *once*, the attacker builds large tables relating to the behaviour of the function in question. In the active phase, the attacker obtains a number of actual data points that she wants to invert, and tries to find a preimage of at least one value using the precomputed tables.

A TMD tradeoff is thus characterised by five parameters; the size of the search space N, the time consumed by the precomputation step P, the amount of memory M used to store the precomputed tables, the time complexity of the active phase T, and the amount of data required D. In the analysis, logarithmic terms are usually ignored. Using these parameters, the Babbage-Golic tradeoff is given by the relations $P = M$, $N = TM$ and $T = D$, while the Biryukov-Shamir tradeoff is given by $P = N/D$, $N^2 = TM^2D^2$, $D^2 \leq T$ [4]. The particular details of how the tradeoffs work and may be implemented in practice are outside the scope of this article.

Modern TMD attacks have been well known to cipher designers for many years, and are taken into account in their designs. In 1995, Babbage stated that as a general rule,

> "if a secret key length of k bits is required, a state size of at least $2k$ bits is desirable" [1]

and this has remained the rule of thumb followed by most new stream ciphers. To provide insight into the motivation for this bound, notice that for a state of size $N = 2^{2k}$, the particular parameter choices $T = M = D = 2^k$ and $T = M = 2^k, D = 2^{k/2}$ are points on the Babbage-Golic and Biryukov-Shamir tradeoff curves, respectively. Since the online complexities T and M are the same as the security level of the cipher, 2^k, the tradeoffs are considered to break even at this point. One should however keep in mind that a (parallel) brute force cracker usually will be more efficient in practice. First, the TMD tradeoffs require huge amounts of memory, keystream and/or precomputation (although precomputation costs are often ignored, or assumed to be amortised over several applications of the online phase), which is not the case with a dedicated cracker. Secondly, the asymptotic analysis ignores all the lower order terms of the tradeoffs, which may turn out to be of significance in practice.

1.2 Description of Grain

In this section we describe the parts of Grain which are relevant to the purposes of this article. For the full specification, we refer to [12]. Grain is bit-oriented stream cipher taking an 80-bit key and a 64-bit IV. The cipher consists of a pair of linked 80-bit shift registers, one linear and one non-linear, whose states at time i is denoted by $\{l_i, \ldots, l_{i+79}\}$ and $\{n_i, \ldots, n_{i+79}\}$.

During key generation, Grain outputs a single bit z_i at each clock cycle, which is computed from 12 of the internal state bits by the equation

$$\begin{aligned} z_i = n_{i+1} + n_{i+2} + n_{i+4} + n_{i+10} + n_{i+31} + n_{i+43} + n_{i+56} \\ + h\big(l_{i+3}, l_{i+25}, l_{i+46}, l_{i+64}, n_{i+63}\big). \end{aligned} \tag{1}$$

In the above, $h(\cdot)$ is a non-linear boolean function of degree 3. Meanwhile, the new LFSR bit is computed using a primitive feedback polynomial $f(x) = 1 + x^{18} + x^{29} + x^{42} + x^{57} + x^{67} + x^{80}$ which yields the update equation

$$l_{i+80} = l_{i+0} + l_{i+13} + l_{i+23} + l_{i+38} + l_{i+51} + l_{i+62}. \tag{2}$$

The NFSR is updated with a non-linear boolean function $g(\cdot)$ of degree six whose output is masked with the low-order LFSR bit:

$$\begin{aligned} n_{i+80} = l_i + g\big(n_i, n_{i+9}, n_{i+14}, n_{i+15}, n_{i+21}, n_{i+28}, \\ n_{i+33}, n_{i+37}, n_{i+45}, n_{i+52}, n_{i+60}, n_{i+62}, n_{i+63}\big). \end{aligned} \tag{3}$$

A notable aspect of Grain is that every bit that enters the registers remains unused for at least 16 clock cycles, since the highest register indices occuring in the state update functions are n_{i+63} and l_{i+64} respectively. This enables implementors to compute up to 16 clocks of the cipher in parallel, greatly speeding up the cipher with only a moderate increase in gate count. Finally, we note that the state update function of Grain is invertible both during keystream generation and key initialisation. This implies that if we recover the state of the cipher at some time t, we can clock it backwards to recover the secret key.

1.3 Related work

The best current attack on Grain v1 is an observation by De Canniere, Küçük and Preneel [6] on sliding properties in key-setup of Grain, which on average allows a brute-force attacker to test keys twice as fast as usual, yielding a worst-case brute force attack complexity of 2^{79} trials. The authors also analyse the key initialisation procedure of Grain, as there has been some concern that it is too brief (even though no specific attack based on this has been found). In [17], Zhang and Wang show that there exist 2^{64} "weak" key/IV pairs for Grain, which lead to the LFSR being zero after the initialisation, and demonstrate that the secret key may be recovered more efficiently than brute force if this occurs.

In the eSTREAM contest itself, there has been relatively little focus on TMD tradeoffs, presumably because the cipher designers have been aware of this class of attacks and have avoided them by design – proposing ciphers whose internal states are not only twice the length of the secret key, but often much larger. However, the TMD tradeoffs proposed by Hong and Kim against MICKEY v1 [16] are of relevance to this paper, as they are very similar to those we obtain for Grain in this article.

Research has also been done on TMD tradeoffs for a different inversion problem, namely that of inverting the one-way function mapping the initial cipher input of key and IV to keystream. At SASC 2008, Dunkelman and Keller [7] proposed a new TMD tradeoff based on chosen IVs which appears to lead to interesting tradeoffs, particularly in cases where the IV is chosen deterministically rather than at random.

2 Time / Memory / Data Cryptanalysis of Grain

Since the 160-bit internal state of Grain is twice the length of the key, the known time / memory / data tradeoff curves can not be applied directly to the function mapping the 160-bit internal state to the subsequent 160 bits of keystream. In this section, we show how better tradeoffs may be obtained.

2.1 Sampling Resistance of Grain

In [4, 5] the authors introduce the concept of BSW-sampling, which can be used to obtain wider choices of tradeoff parameters for the Biryukov-Shamir tradeoff curve. The main idea of BSW-sampling is to find an efficient way to generate and enumerate "special" cipher states, from which some subsequent keystream bits output by cipher are a fixed string (such as a run of consecutive 0-bits). If this can be done for a run of l bits, the *sampling resistance* of the cipher is defined to be $R = 2^{-l}$. This is usually possible (and is not a cause for concern) for small values of l, but leads to improved tradeoff attacks if l is moderately large. We will proceed to show that Grain is vulnerable to BSW-sampling.

Lemma 1. *Given the value of* 133 *particular state bits of Grain and the first* 18 *keystream bits produced from that state, another* 18 *internal state bits may be deduced directly.*

Proof. Recall the form of output function of Grain from Eq. 1, in which the non-linear function $h(\cdot)$ is computed and the result is masked with 7 bits taken from the NFSR state. Our strategy is to exploit the distance between the masking bits n_{i+10} and n_{i+31}, together with the fact that the non-linear feedback of Grain does not affect the output function until the cipher has been clocked 17 times.

At a particular point in time, it is clear that the value of n_{i+10} can be computed directly from the values of z_i and the 11 other state variables occuring in the output equation. Doing this 16 times for $i = 0 \ldots 15$ yields

$$n_{10} = z_0 + n_1 + n_2 + n_4 + n_{31} + n_{43} + n_{56} + \; h(l_3, l_{25}, l_{46}, l_{64}, n_{63}), \tag{4}$$

$$\vdots$$

$$n_{25} = z_{15} + n_{16} + n_{17} + n_{19} + n_{46} + n_{58} + n_{71} + \; h(l_{18}, l_{40}, l_{61}, l_{79}, n_{78}), \tag{5}$$

where at each step the computed NFSR bit has not occured in any of the previous equations. At this point we have fixed the values of 57 NFSR bits and 64 LFSR bits, and deduced 16 of the NFSR bits.

In step 16, the value of l_{80} is used as input to $h(\cdot)$, and we have to fix the values of some additional LFSR bits. Since we are not trying to deduce anything about the LFSR this is not a problem, and we subsequently compute n_{26} without any trouble.

On the 17th step, the value of n_{80} is also used as input to $h(\cdot)$, which means that we also have to take the non-linear feedback into account. In particular, this forces us to fix the value of n_{28} (which is part of the input to the NFSR update function used to compute n_{80} at time $i = 0$). Apart from this complication, we are able to deduce n_{27}.

However, in the 18th step the value of n_{28} was just assigned in the previous step, as are the values of the 6 other linear masking bits. Hence we are not able to keep deducing more bits with this strategy. To sum up, we have recovered the 18 state bits $n_{10} \ldots n_{27}$ using 59 bits of the NFSR state and 74 bits of the LFSR state. The remaining 9 state bits did not occur in any of the equations. □

A slight improvement to the above approach is to only guess enough variables to make the remaining output function linear, and then solve the resulting system of linear equations. By guessing the values of l_{i+3}, l_{i+46} and n_{i+63} in each step, $h(\cdot)$ becomes an affine function in l_{i+25} and l_{i+64}, as these variables do not occur in the same monomials. When the nonlinear feedback begins active, starting at step 17, this can be (temporarily) circumvented by guessing eight of the input values to $g(\cdot)$ and adding a linear constraint on the remaining variables.

In principle, this will allow us to continue for 28 bits of keystream, and obtain a linear system of 39 equations in the 40 remaining (un-guessed) variables. However, as the precise equations obtained are dependent on the particular values of the guessed variables, the procedure can only be applied directly to obtain 21 internal state bits from the first 21 bits of the keystream.

Corollary 1. *The sampling resistance R of Grain is at most* 2^{-21}.

Given that the sampling resistance of Grain is 2^{-21}, we can define[1] a family of one-way functions $\pi_S : \{0,1\}^{139} \to \{0,1\}^{139}$ as follows:

1. Fix a specific function by choosing a 21-bit string S, e.g. 0^{21}.
2. Given an 139-bit input value x, expand it to 160 bits by interpreting it as a partial state value for Grain and computing the remaining 21 bits by the above procedure (treating S as the first 21 bits of keystream).

[1] This construction was first given in [4].

3. Clock Grain 160 steps, generating a 160-bit output string $S|y$.
4. Output y.

It is clear that computing inverses for a particular function π_S is equivalent to the problem of inverting the function mapping 160-bit Grain states to 160-bit keystream segments, restricted to a "special" subset of the cipher states. Furthermore, we are easily able to construct these states, and can compute the function π_S (more or less) as efficiently as the usual state function. Finally, given D bits of actual keystream, the expected number of special states encountered is DR. Hence, we will consider the cost of inverting π_S rather than the full cipher.

2.2 Tradeoff Parameters

The Biryukov-Shamir tradeoff curve for inverting the function π_S can be written as $(RN)^2 = TM^2(RD)^2$, since the domain of the one-way function is reduced by a factor R, but the amount of keystream needed to obtain D data points is increased correspondingly. Hence the real gain from BSW-sampling is that we get a wider choice of parameters on the original tradeoff curve,[2] by relaxing the condition $D^2 \leq T$ to $(RD)^2 \leq T$. Given that we can choose $R = 2^{-21}$ with the sampling scheme proposed, we can use this to increase the total amount of keystream D beyond 2^{40}, and reduce T and M correspondingly. So under the restriction that we want the magnitude of T and M to be less than 2^{80}, some possible parameter choices are given in Fig. 1.

Time (T)	Memory (M)	Keystream (D)	Preprocessing (P)	Comment
2^{62}	2^{77}	2^{52}	2^{108}	Min. time
2^{78}	2^{61}	2^{60}	2^{100}	Min. memory
2^{78}	2^{78}	2^{43}	2^{117}	Min. keystream
2^{74}	2^{65}	2^{58}	2^{102}	
2^{70}	2^{69}	2^{56}	2^{104}	Balanced
2^{66}	2^{73}	2^{54}	2^{106}	
2^{59}	2^{80}	2^{80}	2^{80}	Babbage-Golic

Fig. 1. Possible TMD-tradeoffs for Grain using BSW sampling.

For instance, we see that we can find an attack whose online complexity is bounded by $O(2^{70})$ time and $O(2^{69})$ memory, using $O(2^{56})$ bits of keystream. Regardless of the (un)feasibility of performing this attack in practice, this appears to violate statements given in Section 6.3 of the Grain specifications.

[2] The Babbage-Golic tradeoff is not really helped as much by BSW-sampling, as the amount of keystream needed remains prohibitive. Even if the tradeoff is changed from $N = TM$ to $RN = TM$, the corresponding amount of keystream needed to produce D data points grows to T/R.

> "The cost of time/memory/data tradeoff attacks on stream ciphers is $O(2^{n/2})$, where n is the number of inner states of the stream cipher. To obey the margins set by this attack, n = 160 has been chosen. It is known that stream ciphers with low sampling resistance have tradeoff attacks with fewer table lookups and a wider choice of parameters." [12]

> "Hence the resulting sampling resistance [of Grain] is large, and thus time/memory/data tradeoffs are expected to have complexity not lower than $O(2^{80})$." [12]

When assessing the "cost" of TMD tradeoffs on stream ciphers in the first quote, the authors themselves appear to be neglecting the cost of the precomputation step. Hence it appears that Grain v1 has been designed with a security level of 2^{80} in mind, with respect to the time and memory usage of the *online* phase of a TMD tradeoff attack. Considering only the online phase in our proposed tradeoffs, we see from Table 1 that we can improve significantly on this.

The source of our attack is the observation that Grain has a sampling resistance of at most 2^{-21}, which we in this context do not consider to be particularly "large". It is worth mentioning at this point that the Grain specification does *not* specify any limits to the amount of keystream that may be generated under a single key and IV, so attacks requiring $O(2^{60})$ or more bits of keystream are completely valid in this respect.

2.3 Implications

In fairness, there is little consensus among researchers whether TMD tradeoffs such as those proposed in this paper constitute a real "attack" against Grain. This is both because the precomputation step is more expensive than brute force, and in recognition of the fact that 2^k Grain circuits working in parallel can exhaustively search the keyspace with complexity on the order of 2^{79-k} without the astronomical memory and keystream requirements of a TMD tradeoff.

Worse yet, even if the sampling resistance of Grain can be decreased even further, the Biryukov-Shamir tradeoff will *never* yield a precomputation cost less than $O(2^{80})$. At best, we may optimistically consider a very resourceful adversary that wishes to recover a very large number of different keys, and is able to obtain the required amount of keystream for each key, but even so the amortised cost of this attack does not drop below the cost of brute-force until a quite large amount of keys have been cracked. Admittedly, the assumption that this is a realistic attack scenario is a bit of a stretch.

While some researchers will argue that the result shows that Grain can be broken with an on-line computation which is faster than brute force and that the cost of the precomputation step should be ignored in analysis since it is only performed once, others hold that an "attack" is not relevant as long as the cost of a single (or a few) state recovery attacks remains much greater than brute force. This article does not in any way aim to take sides in or to settle this debate.

However, comparing the situation of Grain and MICKEY v1 [2] seems to be relevant to this discussion. The stream cipher MICKEY is another hardware-oriented cipher, designed by Babbage and Dodd, which was also selected in the final eSTREAM portfolio. In [16], Hong and Kim showed that the sampling resistance of the original version, MICKEY v1, was at most 2^{-27}. As MICKEY v1 also has a state size of 160 bits, the low sampling resistance led to a TMD tradeoff with tradeoffs against that cipher with online complexity $O(2^{67})$ and precomputation cost $O(2^{100})$, whereas the complexity of brute force keysearch would be $O(2^{80})$. The authors write:

> "Owing to the pre-computation complexity larger than exhaustive search of key, some would not view this technically as a *break* of MICKEY. But still, it does show that we cannot treat MICKEY as providing absolutely full 80-bit security." [16]

The response of MICKEY authors Babbage and Dodd is interesting:

> "Some authors seem to ignore precomputation time completely, and consider only online complexity to matter; others would say that an attack requiring overall complexity greater than exhaustive search is of no practical significance. Although we incline more towards the second view, we recognise that some will deem the cipher less than fully secure if such attacks exist." [3]

Partly as a consequence of this TMD tradeoff (although also due to concerns about entropy loss in the state update function), a tweaked version of MICKEY called MICKEY 2.0 was proposed [3]. In the tweaked cipher, the size of the internal state was increased from 160 to 200 bits, which is sufficient to eliminate the possibility of *any* state recovery attack based on BSW-sampling and the Biryukov-Shamir tradeoff curve, since $N^2 = 2^{400} = (2^{80})^5 = TM^2D^2$. However, this comes at a cost of increased gate count, and a decrease in cipher speed.

If one wishes to make Grain secure against the tradeoffs found in this article, what may be done? Looking more closely at our results, the root causes of our obtained tradeoff attack appear to be twofold. The long delay before the non-linear feedback in the NFSR starts affecting the output keystream makes Grain more flexible for implementors, but also makes diffusion slow. We see that our BSW-sampling strategy rapidly fails after nonlinear feedback values start entering the output function. Meanwhile, the minimalistic 160-bit state size of Grain enables us to use the low sampling resistance to obtain good tradeoffs, whereas a 200-bit state would make sampling resistance irrelevant.

Finally, with respect to Grain-128, it can easily be verified that the cipher as specified has a sampling resistance of at most 2^{-22}, by considering the spacing between the linear output masking taps at n_{i+15} and n_{i+36}. A TMD tradeoff similar to the one given for Grain v1 should thus be trivially obtainable, with an attack complexity in the active phase of $O(2^{117})$ using $O(2^{80.5})$ bits of keystream, and a precomputation cost of $O(2^{175.5})$. Due to the 32-step delay between nonlinear feedback and output, and the low degree and simple structure of the nonlinear functions $g(\cdot)$ and $h(\cdot)$, it may be feasible to decrease the sampling resistance towards 2^{-32} or below. The relevance of this observation to the security of Grain-128 will be much the same as for Grain v1.

3 End notes

We have shown that there exists a time/memory/data tradeoff to recover the state of Grain v1 with online time and memory complexity $O(2^{70})$ and $O(2^{69})$ respectively, using $O(2^{56})$ bits of known keystream and a precomputation phase of $O(2^{104})$ steps. Other parameter choices are also available. While the relevance of these attacks can be disputed owing to the huge cost of the precomputation, they still show that, at least under some very special attack scenarios, secret keys can be recovered faster than by exhaustive search of the keyspace. This violates assumptions on the resistance to Grain to TMD tradeoffs made in the official cipher specification, and indicate that Grain's minimalist internal state may in fact be too aggressively specified for some people's tastes.

3.1 Acknowledgements

The author would like to thank Orr Dunkelman, Håvard Raddum and Matthew Parker for early feedback and discussions, as well as the COSIC research group at Katholieke Universiteit Leuven for hosting him while parts of this research was carried out.

References

1. S. Babbage. A space/time tradeoff in exhaustive search attacks on stream ciphers. In *European Convention on Security and Detection*, volume No. 408, 1995.
2. S. Babbage and M. Dodd. The stream cipher MICKEY (version 1). eSTREAM, ECRYPT Stream Cipher Project, 2005.
3. S. Babbage and M. Dodd. The stream cipher MICKEY 2.0. eSTREAM, ECRYPT Stream Cipher Project, 2006. `http://www.ecrypt.eu.org/stream/mickeyp3.html`.
4. A. Biryukov and A. Shamir. Cryptanalytic time/memory/data tradeoffs for stream ciphers. In *Proceedings of ASIACRYPT 2000*, volume 1976 of *Lecture Notes in Computer Science*, pages 1 – 13. Springer–Verlag, 2000.
5. A. Biryukov, A. Shamir, and D. Wagner. Real time cryptanalysis of A5/1 on a PC. In *Proceedings of PKC 2001*, volume 1978 of *Lecture Notes in Computer Science*, pages 37–44. Springer–Verlag, 2001.
6. C. De Cannière, Ö. Küçük, and B. Preneel. Analysis of grain's initialization algorithm. Presented at SASC 2008. `http://www.ecrypt.eu.org/stvl/sasc2008/`.
7. O. Dunkelman and N. Keller. Treatment of the initial value in time-memory-data tradeoff attacks on stream ciphers. Presented at SASC 2008. `http://www.ecrypt.eu.org/stvl/sasc2008/`.
8. eSTREAM, ECRYPT stream cipher project. `http://www.ecrypt.eu.org/stream/`.

9. J. Golic. Cryptanalysis of alleged A5 stream cipher. In *Proceedings of EUROCRYPT 1997*, volume 1233 of *Lecture Notes in Computer Science*, pages 239–255. Springer–Verlag, 1997.
10. T. Good and M. Benaissa. Hardware performance of eSTREAM phase-III stream cipher candidates. Presented at SASC 2008. `http://www.ecrypt.eu.org/stvl/sasc2008/`.
11. M. Hell and T. Johansson. Breaking the F-FCSR-H stream cipher in real time. In *Proceedings of ASIACRYPT 2008*, volume 5350 of *Lecture Notes in Computer Science*, pages 557 – 569. Springer–Verlag, 2008.
12. M. Hell, T. Johansson, A. Maximov, and W. Meier. The Grain family of stream ciphers. *Lecture Notes in Computer Science*, 4986:179–190, 2008. `http://www.ecrypt.eu.org/stream/grainpf.html`.
13. M. Hell, T. Johansson, and W. Meier. Grain – a stream cipher for constrained environments. eSTREAM, ECRYPT Stream Cipher Project, Report 2005/010, 2005.
14. M. Hell, T. Johansson, and W. Meier. A stream cipher proposal: Grain-128. eSTREAM, ECRYPT Stream Cipher Project, 2006. `http://www.ecrypt.eu.org/stream/grainp3.html`.
15. M. Hellman. A cryptanalytic time-memory trade-off. *IEEE Transactions on Information Theory*, 26:401–406, 1980. `http://www-ee.stanford.edu/~hellman/publications/36.pdf`.
16. J. Hong and W.-H. Kim. Tmd-tradeoff and state entropy loss considerations of streamcipher MICKEY. eSTREAM, ECRYPT Stream Cipher Project, Report 2005/055, 2005.
17. Haina Zhang and Xiaoyun Wang. Cryptanalysis of stream cipher Grain family. Cryptology ePrint Archive, Report 2009/109, 2009. `http://eprint.iacr.org/2009/109/`.