Combined Algebraic and Truncated Differential Cryptanalysis on Reduced-round Simon

Nicolas Courtois[1], Theodosis Mourouzis[1], Guangyan Song[1], Pouyan Sepehrdad[2] and Petr Susil[3]
[1]*University College London, London, U.K.*
[2]*Qualcomm Inc., San Diego, CA, U.S.A.*
[3]*École Polytechnique Fédéreale de Lausanne, Lausanne, Switzerland*

Keywords: Lightweight Cryptography, Block Cipher, Feistel, SIMON, Differential Cryptanalysis, Algebraic Cryptanalysis, Truncated Differentials, SAT Solver, Elimlin, Non-linearity, Multiplicative Complexity, Guess-then-determine.

Abstract: Recently, two families of ultra-lightweight block ciphers were proposed, SIMON and SPECK, which come in a variety of block and key sizes (Beaulieu et al., 2013). They are designed to offer excellent performance for hardware and software implementations (Beaulieu et al., 2013; Aysu et al., 2014). In this paper, we study the resistance of SIMON-64/128 with respect to algebraic attacks. Its round function has very low Multiplicative Complexity (MC) (Boyar et al., 2000; Boyar and Peralta, 2010) and very low non-linearity (Boyar et al., 2013; Courtois et al., 2011) since the only non-linear component is the bitwise multiplication operation. Such ciphers are expected to be very good candidates to be broken by algebraic attacks and combinations with truncated differentials (additional work by the same authors). We algebraically encode the cipher and then using guess-then-determine techniques, we try to solve the underlying system using either a SAT solver (Bard et al., 2007) or by ElimLin algorithm (Courtois et al., 2012b). We consider several settings where P-C pairs that satisfy certain properties are available, such as low Hamming distance or follow a strong truncated differential property (Knudsen, 1995). We manage to break faster than brute force up to 10(/44) rounds for most cases we have tried. Surprisingly, no key guessing is required if pairs which satisfy a strong truncated differential property are available. This reflects the power of combining truncated differentials with algebraic attacks in ciphers of low non-linearity and shows that such ciphers require a large number of rounds to be secure.

1 INTRODUCTION

Nowadays, due to the continuously growing impact of mobile phones, smart cards, RFID tags, sensor networks, there is a great demand to provide security and design cryptographic algorithms which are suitable and can be efficiently implemented in very resource-constrained devices. The area of cryptography which studies the design and the security of such lightweight cryptographic primitives, called lightweight cryptography, is rapidly evolving and becoming more and more important.

These lightweight cryptographic primitives are designed to be efficient (in both hardware and software) when limited hardware resources are available and at the same time to guarantee a desired level of security. The design of such primitives is a great challenge and can be seen as a non-trivial optimization problem, where several trade-offs are taken into account. They need to maintain a reasonable balance between security, efficient software and hardware implementation and very low overall cost with respect to several meaningful metrics (power consumption, energy consumption, size of the circuit (Courtois et al., 2011; Boyar and Peralta, 2010; Boyar et al., 2000)).

In July 2013, a team from the NSA has proposed two new families of particularly lightweight block ciphers, SIMON and SPECK, both coming in a variety of widths and key sizes (Beaulieu et al., 2013). We use a basic reference implementation of both ciphers which can be found in (Courtois et al., 2014), as well as the generator of algebraic equations to be used in algebraic attacks.

However, no advanced analysis of the security of the ciphers was discussed. In the same paper (Beaulieu et al., 2013), they briefly said that SIMON and SPECK were designed to provide security against traditional adversaries who can adaptively encrypt

Courtois N., Mourouzis T., Song G., Sepehrdad P. and Susil P..
Combined Algebraic and Truncated Differential Cryptanalysis on Reduced-round Simon.
DOI: 10.5220/0005064903990404
In *Proceedings of the 11th International Conference on Security and Cryptography* (SECRYPT-2014), pages 399-404
ISBN: 978-989-758-045-1

and decrypt large amounts of data and some attention was paid so that there are no related-key attacks. No analysis against common attacks such as linear or differential cryptanalysis was presented and the task of analyzing the resistance of the ciphers against known attacks was left to the academic community. Immediately after the release of the specifications we had the first attempts of cryptanalysis against differential, linear and rotational cryptanalysis (Farzaneh et al., 2013; Alkhzaimi and Lauridsen, 2013). However, since SIMON is a cipher of exceptionally low MC and as a result of low non-linearity it is an ideal candidate for algebraic attacks and combinations of algebraic attacks with truncated differential cryptanalysis.

Contribution and Outline. In this paper, we study SIMON-64/128 against algebraic attacks and combinations of algebraic attacks with truncated differential cryptanalysis. Our aim is to use the very rich algebraic structure with additional data provided (e.g. pairs $\{(P,P'),(C,C')\}$ which follow a certain highly probable truncated differential property) in order to solve the underlying multivariate system of equations. We attempt to solve the system by either using a SAT solver (after converting the system to its corresponding CNF-SAT form (Bard et al., 2007)) or by ElimLin algorithm (Boyar et al., 2013; Courtois et al., 2011; Courtois et al., 2013). We are able to break up to 10 (/44) rounds of the cipher using a SAT solver and usual guess-then-determine techniques. Surprisingly, in most cases we are able to obtain the key without guessing any key bits (Susil et al., 2014) when truncated differentials are used. This is a very remarkable results since one of the biggest difficulties in software algebraic cryptanalysis is to know how do experimental attacks scale up for larger numbers of rounds. This is possibly due to the very low non-linearity of the cipher and suggests that it worths studying a specific strategy for P-C pairs which have certain structure and decrease even more the non-linearity of the system by introducing more linear equations (e.g. truncated differential properties) until the key can be obtained even for more number of rounds. We discuss in details our results in Section 4.

In our attacks, we study the following three settings:

1. **RP/RC.** (Random Plaintexts/Random Ciphertexts): We assume that random plaintext-ciphertext (P-C) pairs are available
2. **SP/RC.** (Similar Plaintexts/Random Ciphertexts): We assume that random P-C pairs such that the plaintexts differ by very few bits are available (low Hamming distance)
3. **SP/SC.** (Similar Plaintexts/Similar Ciphertexts): We assume that random P-C pairs which follow a highly probably truncated differential property are available.

In Section 2 we discuss the specification of SIMON-64/128 version. In Section 3, we provide some introduction regarding algebraic cryptanalysis, MC and the ElimLin algorithm. We discuss software algebraic cryptanalysis as a 2-step process, where in the first step we algebraically encode the problem and then we aim to solve the underlying system of equations using available software solver.

2 GENERAL DESCRIPTION OF SIMON

SIMON is a family of lightweight block ciphers with the aim to have optimal hardware performance (Beaulieu et al., 2013). It follows the classical Feistel design paradigm, operating on two n-bit halves in each round and thus the general block size is $2n$. The SIMON block cipher with an n-bit word is denoted by Simon-$2n$, where $n = 16, 24, 32, 48$ or 64 and if it uses an m-word key (equivalently mn-bit key) we denote it as Simon-$2n/mn$. In this paper, we study the variant of SIMON with $n = 32$ and $m = 4$ (i.e. 128-bit key).

Each round of SIMON applies a non-linear, non-bijective (and as a result non-invertible) function

$$F : GF(2)^n \rightarrow GF(2)^n \tag{1}$$

to the left half of the state which is repeated for 44 rounds. The operations used are as follows:

1. bitwise XOR, $\oplus$
2. bitwise AND,
3. left circular shift, S^j by j bits.

We denote the input to the i-th round by $L^{i-1}||R^{i-1}$ and in each round the left word L^{i-1} is used as input to the round function F defined by,

$$F(L^{i-1}) = (L^{i-1} <<< 1) \wedge (L^{i-1} <<< 8) \oplus (L^{i-1} <<< 2) \tag{2}$$

Then, the next state $L^i||R^i$ is computed as follows (cf. Fig. 1),

$$L^i = R^{i-1} \oplus F(L^{i-1}) \oplus K^{i-1} \tag{3}$$

$$R^i = L^{i-1} \tag{4}$$

The output of the last round is the ciphertext.

The key schedule of SIMON is based on an LSFR-like procedure, where the nm-bits of the key are used to generate the keys $K_0, K_1, ..., K_{r-1}$ to be used in each

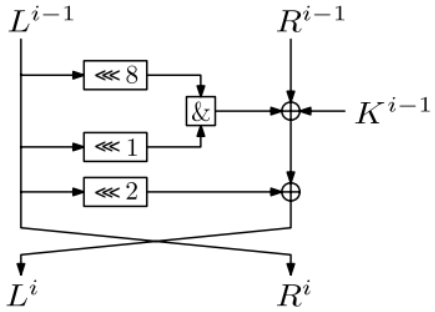

Figure 1: The round function of SIMON.

round. There are three different key schedule procedures depending on the number of words that the secret key consists of ($m = 2, 3, 4$).

At the beginning, the first m words $K^0, K^1, ..., K^{m-1}$ are initialized with the secret key, while the remaining are generated by the LSFR-like construction. For the variant of our interest, where $m = 4$, the remaining keys are generated in the following way:

$$Y = K^{i+1} \oplus (K^{i+3} >>> 3) \quad (5)$$

$$K^{i+4} = K^i \oplus Y \oplus (Y >>> 1) \oplus c \oplus (z_j)_i \quad (6)$$

The constant $c = 0xff...fc$ is used for preventing slide attacks and attacks exploiting rotational symmetries (Beaulieu et al., 2013). In addition, the generated subkeys are xored with a bit $(z_j)_i$, that denotes the i-th bit from the one of the five constant sequences $z_0, ..., z_4$. These sequences are defined in (Beaulieu et al., 2013) and for our variant we use z_3..

3 ALGEBRAIC ATTACKS AND METHODS FOR SOLVING STAGE

Claude Shannon, has once suggested that the security of a cipher should be related to the difficulty of solving the underlying system of equations and deriving the key (Shannon, 1949). Initially, this was not understood and since then only the statistical and *local* aspects of the ciphers were studied. This *global* approach was not considered. Probabilistic attacks very often require huge amount of data, while algebraic attacks can be attempted with very little data. In general, an algebraic attack can be a form of *known-plaintext attack* which consists of the following two basic steps; the **modeling step** where the cipher is described as a multivariate system of polynomial equations and the **solving stage** where we solve the system.

The modeling step is not trivial and one simple method is to follow closely a hardware implementation of the cipher as a circuit (Courtois and Bard, 2007). The most challenging part is the second part where some extra assumptions and information are needed until the system is solvable. Such method is guess-then-determine techniques, where some random bits (or carefully chosen bits depending on the structure) of the key are guessed and then try to solve for the rest key bits.

Several methods for solving the underlying system of equations are known. One method is to compute the corresponding CNF-SAT form of the problem and then attempt to derive the solution using a SAT solver software (Bard et al., 2007). The advantage of such technique is that SAT solvers can perform reasonably well and do not require a lot of memory as in case of Gröbner basis-based techniques (Faugere, 1999). The only disadvantage is the unpredictability of its complexity.

Another method, is to use ElimLin algorithm (Courtois et al., 2012b). ElimLin stands for **Elim**inate **Lin**ear and it is a simple algorithm for solving polynomial systems of multivariate equations over small finite fields and was initially proposed as a single tool by Courtois to attack DES (Courtois and Bard, 2007). It is also known as "inter-reduction" step in all major algebra systems.

Its main aim is to reveal some hidden linear equations existing in the ideal generated by the system of polynomials. ElimLin is composed of two sequential stages, as follows:

- **Gaussian Elimination:** Discover all the linear equations in the linear span of initial equations.
- **Substitution:** Variables are iteratively eliminated in the whole system based on linear equations found until there is no linear equation left.

This method is iterated until no linear equation is obtained in the linear span of the system. Intuitively, ElimLin seems to work better in cases where there is low non-linearity since this implies the existence of more linear equations. MC is another notion of non-linearity (Boyar et al., 2013; Courtois et al., 2013) and possibly such method may work sufficiently well in cryptographic primitives of low MC.

In this paper, we apply both techniques for solving the underlying system of equations that describes SIMON cipher. In order to introduce more linearity to the system, we use either P-C pairs with plaintexts of low Hamming distance which are used to eliminate many variables in the initial equations or pairs which

satisfy certain truncated differential properties. Combining differential and algebraic attacks is like reducing the number of non-trivial variables in the system and thus increasing the probability of solving the system by computer algebra tools. For example, a truncated differential mask [0000022200000080], with four active bits for the input variables, is equivalent to adding 60 linear equations of the form $P_i \oplus P_j = 0$.

4 ALGEBRAIC CRYPTANALYSIS OF SIMON

We evaluated the security of SIMON against algebraic attacks under the following three settings (cf. Fig. 2), where S=Similar and R=Random.

Setting 1 is the simplest setting of understanding how many rounds of SIMON can be broken by simple guess-then-determine techniques, assuming availability of a few P-C pairs. This setting help us to understand the maximum number of rounds we can break by guessing as few as possible key bits and using as few as possible P-C pairs. It a non-trivial step in order to set the benchmark for attacking more number of rounds.

An attack using Setting 2 is a form of known-plaintext attack. Setting 2 requires P-C pairs with plaintexts of low Hamming distance such that many variables are eliminated in the first few rounds.

Lastly, Setting 3 requres P-C pairs

$$\{(P_1,C_1),(P_2,C_2),...,(P_n,C_n)\}$$

such that $P_i \oplus P_j \in \Delta P$ and $C_i \oplus C_j \in \Delta C$, for all $1 \leq i,j \leq n$ and some truncated differential masks $\Delta P, \Delta C$ of low Hamming weight which holds with comparatively high probability. In our attacks we always use 2 pairs which satisfy a given truncated differential property and then more P-C pairs are generated by using the first 2 plaintexts and computing the encryptions of new plaintexts which have small Hamming distance from the first ones. The difference from Setting 2 is that in this case we also eliminate variables from the last rounds of the cipher, expecting that the system is even more easier to solve.

We run experiments using SAT solvers and ElimLin Algorithm on a machine with the following characteristics, CPU: Intel i7-3520m 2.9GHz, RAM: 4G and OS: 64-bit Windows 8.

In all of our attacks we use the best 8 and 10 round truncated differential property $\Delta = [0000022200000080]$ (with 4 active bits and propagates with probability $2^{-20.51}$) and $[0000002200000080] \rightarrow [002eff9a00022e4c]$ (probability $2^{-16.96}$) respectively. More details about our discovery method will be published in future papers.

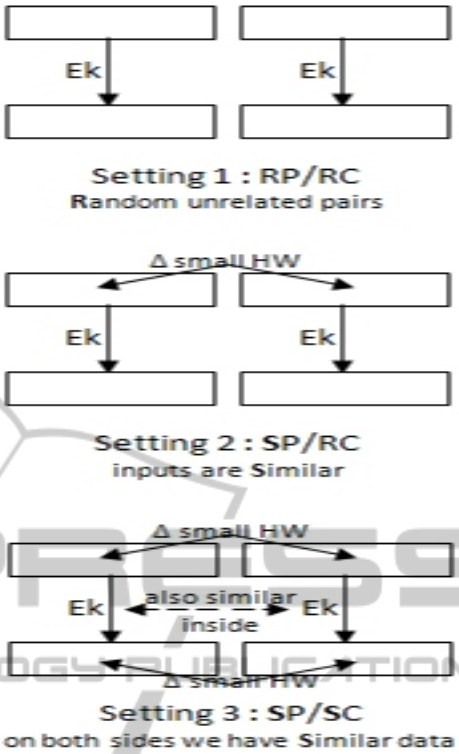

Figure 2: Our three attack scenarios.

4.1 ALGEBRAIC ATTACKS BASED ON SAT SOLVERS

Table 1 presents the best result obtained for several rounds using a SAT solver. The average time (in seconds) taken $T_{average}$ to solve the underlying problem by a SAT solver is presented, while the time complexity C_T (in terms of SIMON encryptions) is computed by the following formulae,

$$C_T = 2^k \times 2^{log_2(T_{average})}, \quad (7)$$

where k is the number of bits we guess initially.

From Table 1, we observe that up to 9 rounds we can solve the underlying system of equations without guessing any key bits initially provided we have Settings 2 and 3. Setting 3 seems to be slightly better up to 9 rounds.

Moreover, assuming Setting 3 we can break 10 rounds by guessing 70 bits of the key initially with time complexity $2^{98.79}$ encryptions. Note that in SP/SC setting we always generate two P-C pair which s satisfy the truncated differential property and the rest pairs are generated from them assuming low Hamming Distance in plaintexts.

Table 1: Best results obtained by a SAT solver. n, s, h stand for number of variables, sparsity and hardness respectively and m the number of equations. We define hardness as a number h such that h^n is the running time, where n is the number of variables. It is known that $h \leq 1.47$ for 4-SAT problems (cf. Table 1 in (Semaev and Mikus, 2010)).

#(Rounds)	k	#(P-C)	$T_{average}$(s)	C_T	Setting	n	$x = \frac{m}{n}$	s	h
8	45	6	207.0	$2^{27.78}$	RP/RC	8576	6.51	4.28	1.0032
8	75	2	156.6	$2^{102.4}$	SP/RC	2944	6.34	4.27	1.0092
8	0	6	12.8	$2^{23.8}$	SP/RC	8576	6.51	4.28	1.0029
9	0	6	222.5	$2^{27.9}$	SP/RC	9536	6.70	4.31	1.0029
9	0	7	94.7	$2^{26.6}$	SP/RC	11104	6.71	4.31	1.0024
10	90	8	346.0	$2^{118.5}$	SP/RC	13952	6.90	4.32	1.002
8	0	6	11.2	$2^{23.6}$	SP/SC	8576	6.55	4.26	1.0028
9	0	7	18.56	$2^{24.30}$	SP/SC	9536	6.70	4.31	1.0026
10	70	10	417.73	$2^{98.79}$	SP/SC	17408	6.88	4.31	1.0022

4.2 Algebraic Attacks Using ElimLin Algorithm

Table 2 presents the results using the ElimLin algorithm.

Table 2: Best results obtained by a ElimLin Algorithm.

#(Rounds)	k	#(P-C)	$T_{average}$(s)	C_T	Setting
8	0	6	824.4	$2^{29.8}$	SP/RC
8	0	6	583.2	$2^{29.3}$	SP/SC

We have been able to break up to 8 rounds in Setting 3 without guessing any key bits initially. The best attack we have obtained is of time complexity $2^{29.3}$ encryptions against 8 rounds of SIMON. Adding pairs which follow a strong truncated differential property is equivalent to adding linear equations in the system and this is exploited by the ElimLin algorithm. An immediate improvement is to use additional intermediate truncated differences and this will be our future work.

5 CONCLUSIONS AND FUTURE RESEARCH

Nowadays, there is a great demand to design lightweight cryptographic primitives which are suitable to be implemented in very resource-constrained devices. The research community has proposed several lightweight cryptographic primitives and recently two families of lightweight block ciphers were proposed, Simon and Speck (Beaulieu et al., 2013).

In (Beaulieu et al., 2013) only the implementation aspects of the ciphers are discussed but immediately after their release, a few attacks against reduced-round versions of the ciphers were discovered; we have mainly differential attacks (Farzaneh et al., 2013; Alkhzaimi and Lauridsen, 2013) and attacks using impossible differentials (Farzaneh et al., 2013) (cf. Table 3).

In this paper, we studied the security of Simon-64/128 cipher against algebraic attacks and algebraic-differential attacks. We have combined two powerful cryptanalytic techniques: truncated differential cryptanalysis and software algebraic cryptanalysis. To the best of our knowledge we are the first to show that such a combination is powerful enough to break up to 10 rounds of a block cipher without guessing any key bits. How important is it to be able to break ciphers without guessing any key bits (Susil et al., 2014) ?

One of the biggest difficulties in software algebraic cryptanalysis is to know how do experimental attacks scale up for larger numbers of rounds. A pessimistic version says that there is a combinatorial explosion of dependencies with respect to key variables and there is no hope to break many rounds much faster than brute force (Courtois et al., 2012a), even though it might be possible to break more rounds very slightly faster than brute force. An optimistic version becomes more plausible when we exhibit poly-time like attacks in which there is no guessing of key variables whatsoever, all variables are determined. Such algebraic attacks are expected to scale up much better for larger numbers of rounds.

Table 3: State-of-art regarding cryptanalysis of Simon-64/128.

Authors	Rounds Attacked	Type	Time	Data	Memory
(Farzaneh et al., 2013)	24/44	Diff	$2^{58.427}$	$2^{62.012}$	2^{32}
(Farzaneh et al., 2013)	16/44	Imp-Diff	$2^{91.986}$	$2^{65.248}$	$2^{60.203}$
(Alkhzaimi and Lauridsen, 2013)	26/44	Diff	$2^{94.0}$	$2^{63.0}$	$2^{31.0}$
(Biryukov et al., 2014)	26/44	Diff	$2^{121.0}$	$2^{63.0}$	$2^{31.0}$
This paper	9/44	Alg	$2^{29.8}$	$2^{2.59}$	negl.
This paper	10/44	Alg	$2^{118.5}$	$2^{3.0}$	negl.
This paper	10/44	Trunc-Diff-Alg	$2^{98.79}$	2^{17}	negl.

REFERENCES

Alkhzaimi, H. and Lauridsen, M. (2013). Differential and linear cryptanalysis of reduced-round simon. In *Cryptology ePrint Archive, Report 2013/543*.

Aysu, A., Gulcan, E., and Schaumont, P. (2014). Simon says, break the area records for symmetric key block ciphers on fpgas. In *Cryptology ePrint Archive, Report 2014/237*.

Bard, G., Courtois, N., and Jefferson, C. (2007). Efficient methods for conversion and solution of sparse systems of low-degree multivariate polynomials over gf(2) via sat-solvers.

Beaulieu, R., Shors, D., Smith, J., Treatman-Clark, S., Weeks, B., and Wingers, L. (2013). The simon and speck families of lightweight block ciphers. In *Cryptology ePrint Archive, Report 2013/404*.

Biryukov, A., A. Roy, A., and Velichkov, V. (2014). Differential analysis of block ciphers simon and speck. In *21st International Workshop on Fast Software Encryption, FSE 2014*.

Boyar, J., Find, M., and Peralta, R. (2013). Four measures of nonlinearity. In *In Algorithms and Complexity, pp. 61-72. Springer Berlin Heidelberg*.

Boyar, J. and Peralta, R. (2010). A new combinational logic minimization technique with applications to cryptology.

Boyar, J., Peralta, R., and Pochuev, D. (2000). On the multiplicative complexity of boolean functions over the basis. In *Theoretical Computer Science 235, no. 1, pp. 43-57*.

Courtois, N. and Bard, G. (2007). Algebraic cryptanalysis of the data encryption standard. In *In IMA Int. Conf. volume 4887, Springer*.

Courtois, N., Gawinecki, J., and Song, G. (2012a). Contradiction immunity and guess-then-determine attacks on gost. In *In Tatra Mountains Mathematic Publications, Vol. 53 no. 3, pp. 65-79*.

Courtois, N., Hulme, D., and Mourouzis, T. (2011). Solving circuit optimisation problems in cryptography and cryptanalysis. In *In electronic proceedings of 2nd IMA Conference Mathematics in Defence 2011*.

Courtois, N., Mourouzis, T., and Hulme, D. (2013). Exact logic minimization and multiplicative complexity of concrete algebraic and cryptographic circuits. In *To Appear in IARIA Journal: IntSys13v6n34*.

Courtois, N., Mourouzis, T., and Song, G. (2014). Reference implementation of simon and speck and a basic generator of equations - https://github.com/gsonghashrate/simonspeck/.

Courtois, N., Sepehrdad, P., Susil, P., and Vaudenay, S. (2012b). Elimlin algorithm revisited. In *Fast Software Encryption, pp. 306-325, Springer Berlin Heidelberg*.

Farzaneh, A., List, E., Lucks, S., and Wenzel, J. (2013). Differential and linear cryptanalysis of reduced-round simon. In *Cryptology ePrint Archive, Report 2013/526*.

Faugere, J.-C. (1999). A new efficient algorithm for computing grobner bases (f4). In *Journal of pure and applied Algebra, Vol. 139, pp. 61-88*.

Knudsen, L. (1995). Truncated and higher order differentials. In *In Fast Software Encryption, pp. 196-211, Springer Berlin Heidelberg*.

Semaev, I. and Mikus, M. (2010). Methods to solve algebraic equations in cryptanalysis. In *In Tatra Mountains Mathematic Publications, Vol. 45, pp. 107-136*.

Shannon, C. (1949). Communication theory of secrecy systems. In *Bell System Technical Journal 28*.

Susil, P., Sepehrdad, P., and Vaudenay, S. (2014). On selection of samples in algebraic attacks and a new technique to find hidden low degree equations. In *ACISP*.