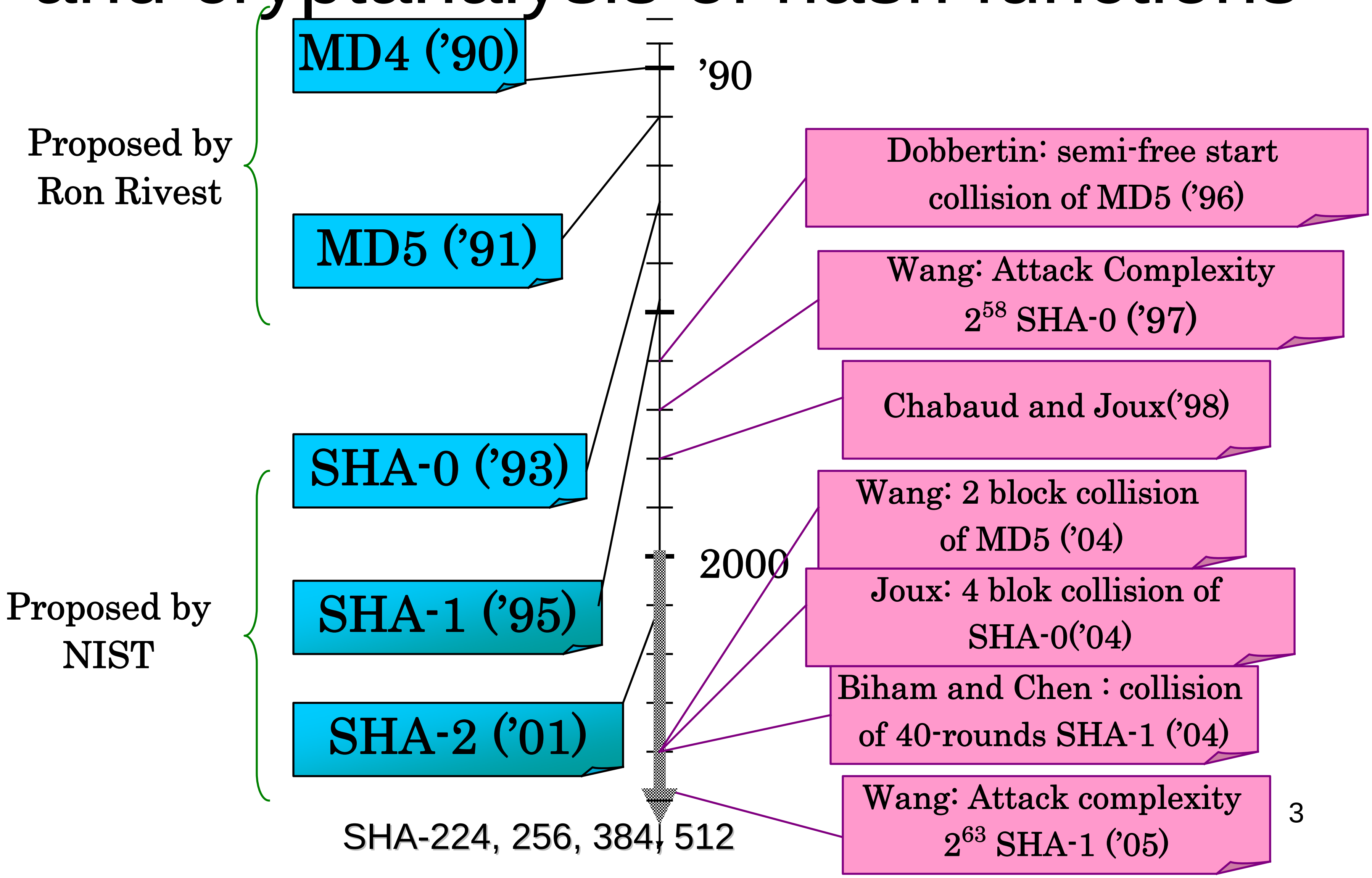
Gröbner Basis Based Cryptanalysis of SHA-1

Makoto Sugita
IPA Security Center
Joint work with Mitsuru Kawazoe (Osaka Prefecture university) and Hideki Imai (Chuo University and RCIS, AIST)

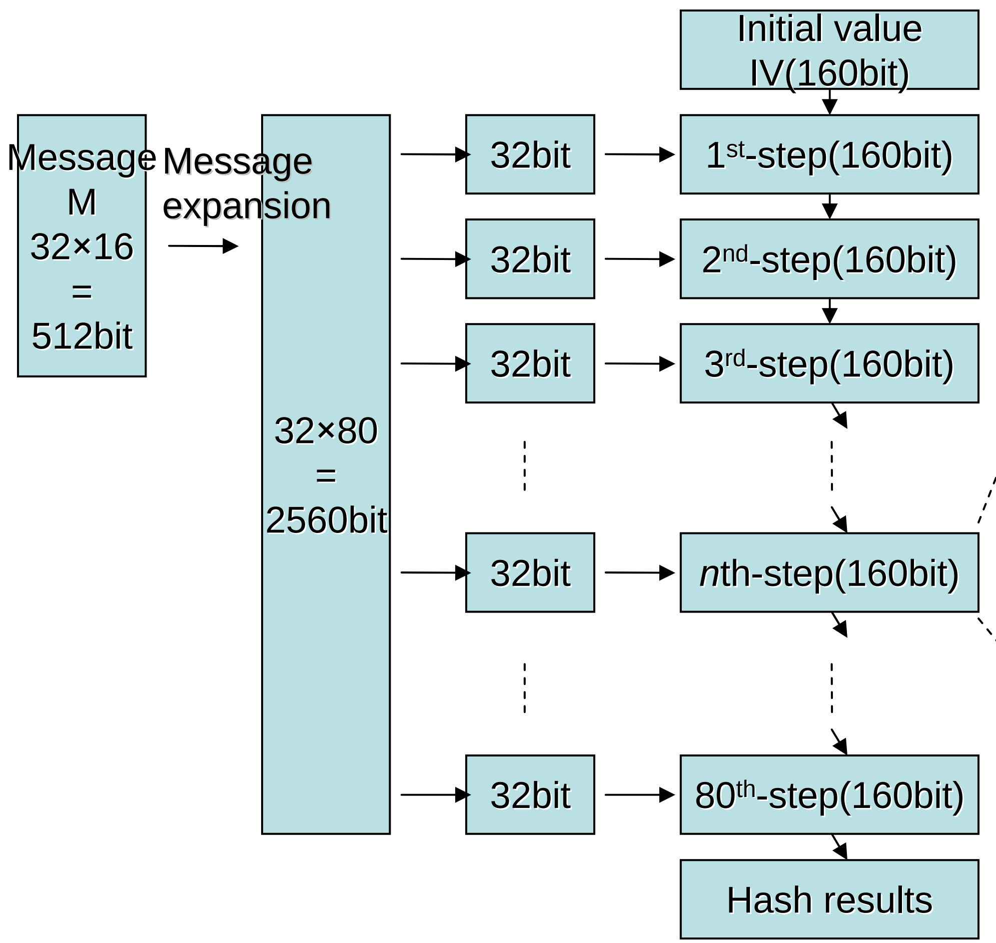
Outline

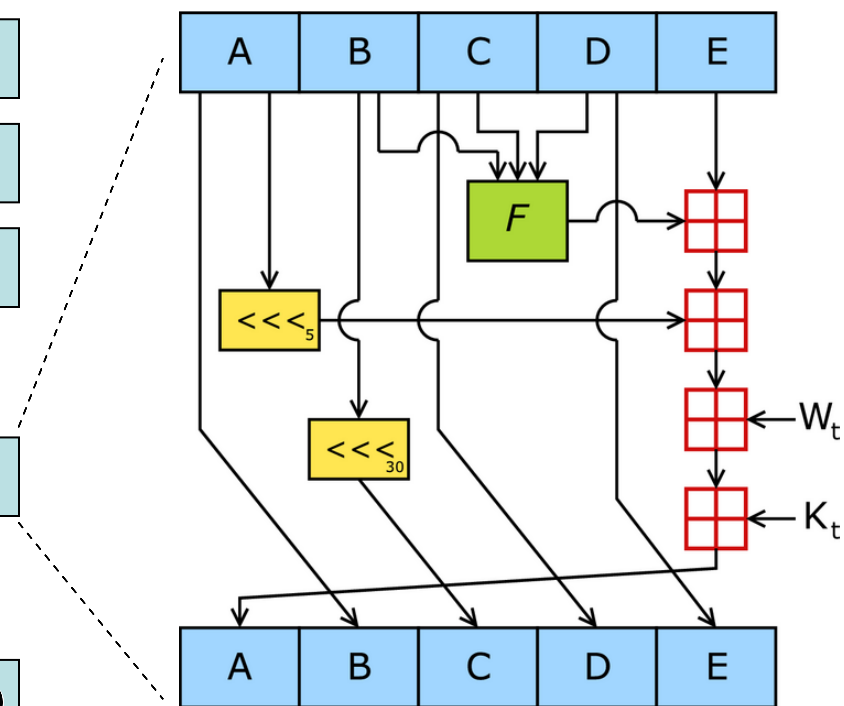
- Introduction
- Wang's method
- Our method - Gröbner basis based method
- Gröbner basis based cryptanalysis of 58-round SHA-1
- Gröbner basis based cryptanalysis of full-round SHA-1
- Conclusion

A history of hash function proposals and cryptanalysis of hash functions



Structure of hash function SHA-1





A, B, C, D,E :32-bit words of the state

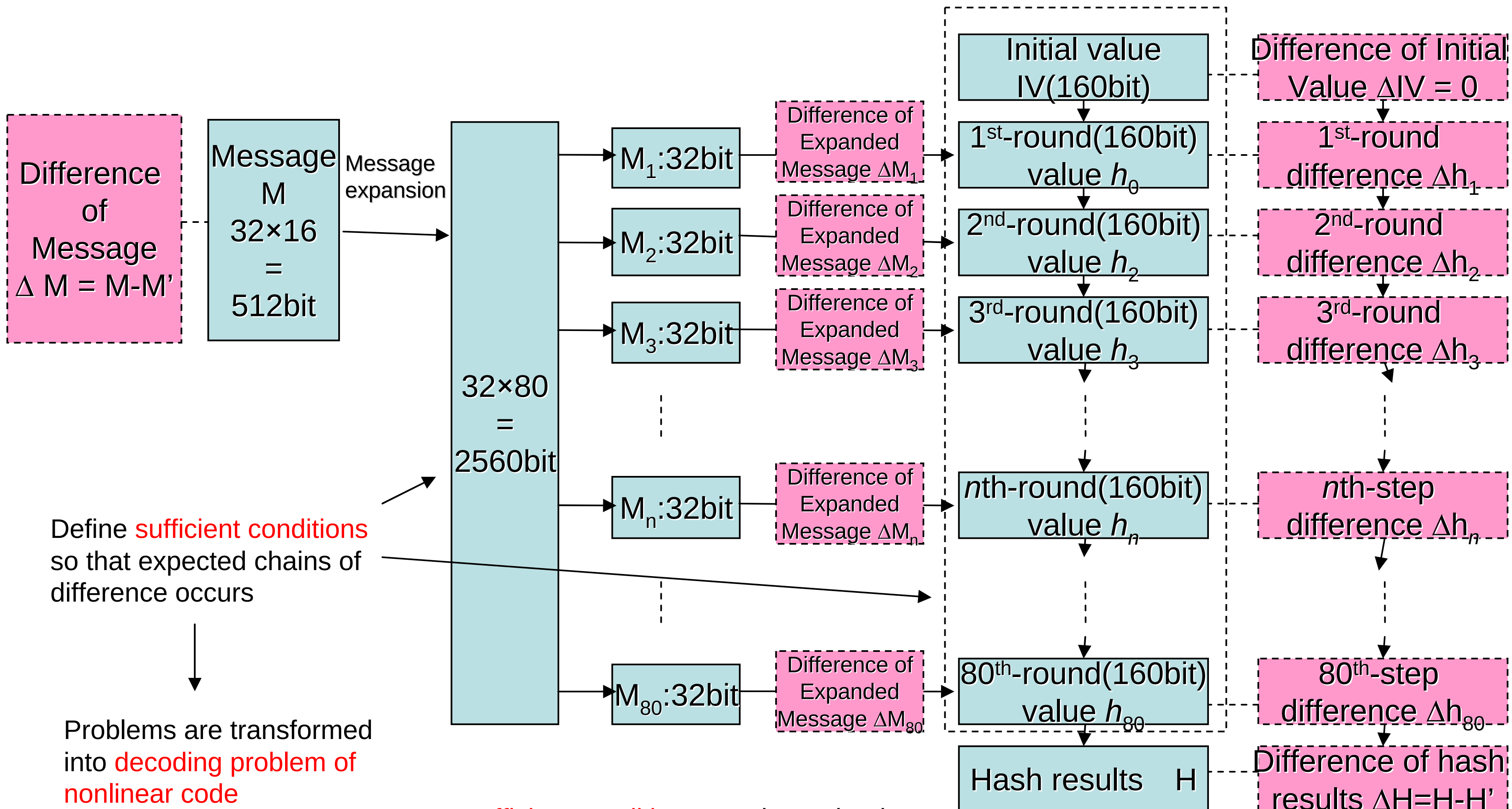
F : nonlinear function

$<<<_s$: left bit rotation by *s* places;

: addition modulo 2^{32}.

Kt :constant.

Differential cryptanalysis against Hash functions



sufficient conditions are determined depending on differential values (disturbance vector)

$\Delta H=0 \Rightarrow$ collision: H = H'

Wang's attack

Outline of the attack.

- Find differential paths – characteristics (difference for subtractions modular 2^{32})
- Determine certain sufficient conditions
- For randomly chosen M, apply the message modification techniques
- However, not all information is published
  - How to find such differential path (disturbance vector)?
    - Candidates are too many
  - How to determine sufficient conditions?
  - What is multi-message modification?
    - Details are unpublished

Sufficient condition and message modification techniques by Wang

chaining variable	conditions on bits 32 – 25	24 – 17	16 – 9	8 – 1
a_1	a00-----	--------	1-----aa	1-0a11aa
a_2	01110---	------1-	0aaa-0--	011-001-
a_3	0-100---	-0-aaa0-	--0111--	01110-01
a_4	10010---	a1---011	10011010	10011-10
a_5	001a0---	--01-000	10001111	-010-11-
a_6	1-0-0011	1-1001-0	111011-1	a10-00a-
a_7	0---1011	1a0111--	101--010	-10-11-0
a_8	-01---10	000000aa	001aa111	---01-1-
a_9	-00-----	10001000	0000000-	---11-1-
a_{10}	0-------	1111111-	11100000	0-----0-
a_{11}	--------	------10	11111101	1-a--0--
a_{12}	0-------	--------	--------	10--11--
a_{13}	--------	--------	--------	11----10
a_{14}	-0------	--------	--------	----0-1-
a_{15}	10------	--------	--------	----1-0-
a_{16}	--1-----	--------	--------	----0-0-
a_{17}	0-0-----	--------	--------	------1-
a_{18}	--1-----	--------	--------	----a---
a_{19}	--b-----	--------	--------	------0-
a_{20}	--------	--------	--------	-----a--
a_{21}	--------	--------	--------	-------1

Table 10 A set of sufficient conditions on a_i for the 80-step differential path given in Table 9. b denote the condition $a_{19,30} = a_{18,32}$

Method for determining sufficient conditions is unpublished

Many details are not public!!

1. How to find the differentials?
2. How to determine sufficient conditions on a_i?
3. What are the details of message modification technique?

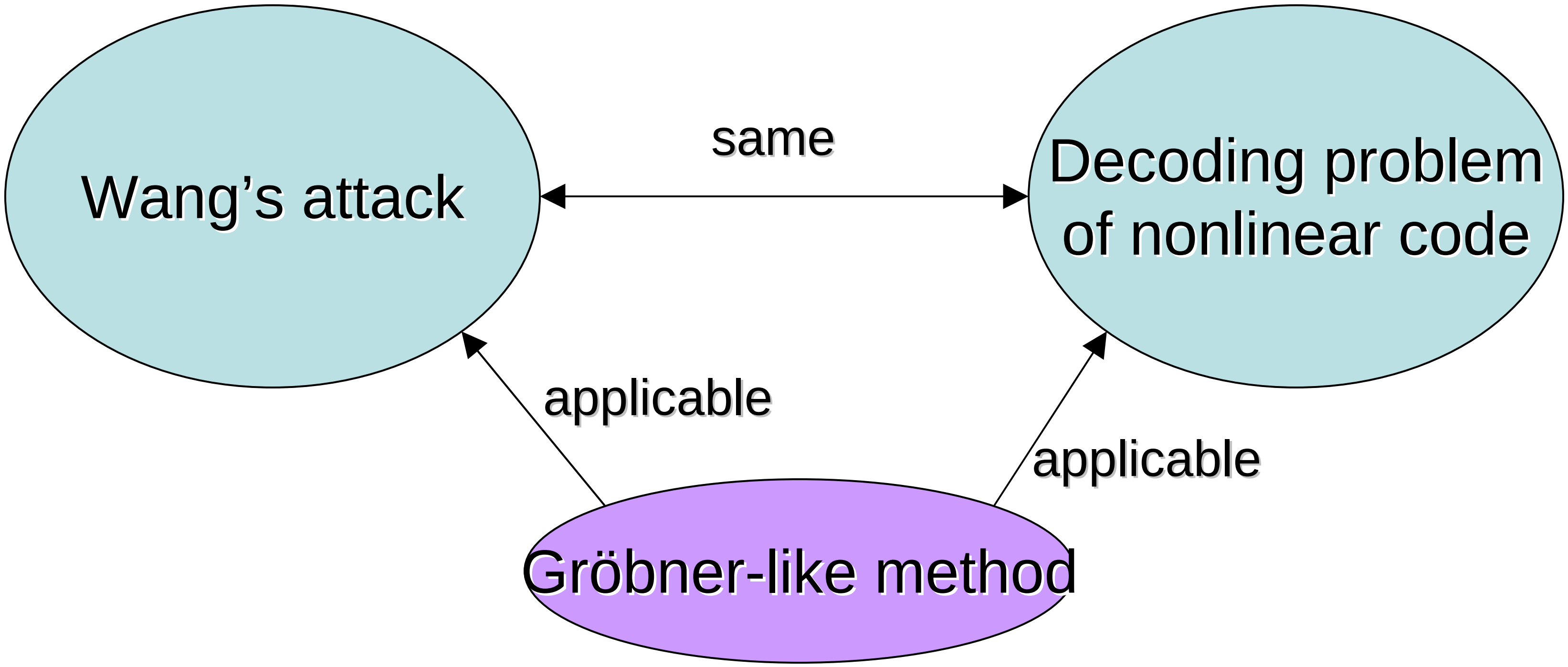
=>

We have clarified 2 and 3, and partially 1

Our Contribution:

- Developing the searching method for 'good' message differentials
- Developing the method to determine sufficient conditions
- Developing new multi-message modification technique
  - Proposal of a novel message modification technique employing the Gröbner basis based method

Wang’s attack, nonlinear code and Gröbner basis



- Wang’s attack can be considered as decoding problem of nonlinear code.

Wang's attack and nonlinear code

- Wang's attack is decoding a nonlinear code $\{a_i, m_i\}$ in $GF(2)^{32x80x2}$.
  - Satisfying sufficient conditions
  - Satisfying nonlinear relations between *a* and *m*

$$m_i = (m_{i-3} \oplus m_{i-8} \oplus m_{i-14} \oplus m_{i-16}) \lll 1$$

for $i = 16, \cdots, 79$, where $x \lll n$ denotes n-bit left rotation of x. Using expanded messages, for $i = 1, 2, \cdots, 80$,

$$\begin{aligned} a_i &= (a_{i-1} \lll 5) + f_i(b_{i-1}, c_{i-1}, d_{i-1}) + e_{i-1} + m_{i-1} + k_i \\ b_i &= a_{i-1} \\ c_i &= b_{i-1} \lll 30 \\ d_i &= c_{i-1} \\ e_i &= d_{i-1} \end{aligned}$$

where initial chaining value $IV = (a_0, b_0, c_0, d_0, e_0)$ is $(0x67452301, 0xefcdab89, 0x98badcfe, 0x10325476, 0xc3d2e1f0)$.

How to decode nonlinear code?

- A general method
  - Gröbner bases based algorithm
- Difficult to calculate Gröbner basis directly:
  - System of equations is very complex
- How to decode?
  - Employ Gröbner basis based method
  - Employ techniques of error correcting code
  - Note: Nonlinear relations between *a* and *m* can be linearly approximated

How to find disturbance vector and construct differentials?

- See our preprint. After that, some better methods have already been published by other teams.
- We recently proposed a new non-probabilistic method to construct differentials using `Rail Differential` in SCIS2007 in Japan

	Δm	Δm w/o carry
i = 0	a8000041	a8000041
i = 1	8000001c	80000014
i = 2	28000042	28000042
i = 3	70000042	10000042
i = 4	38000013	28000011
i = 5	b8000020	88000020
i = 6	a0000000	a0000000
i = 7	e0000032	20000012
i = 8	a0000043	a0000041
i = 9	20000048	20000048
i = 10	a0000040	a0000040
i = 11	f0000042	10000042
i = 12	90000010	90000010
i = 13	10000040	10000040
i = 14	a0000003	a0000003
i = 15	20000030	20000030
i = 16	60000000	60000000
i = 17	e000002a	e000002a
i = 18	20000043	20000043
i = 19	b0000040	b0000040
i = 20	d0000053	d0000053
i = 21	d0000022	d0000022
i = 22	20000000	20000000
i = 23	60000032	60000032
i = 24	60000043	60000043
i = 25	20000040	20000040
i = 26	e0000042	e0000042
i = 27	60000002	60000002
i = 28	80000001	80000001
i = 29	00000020	00000020
i = 30	00000003	00000003
i = 31	40000052	40000052
i = 32	40000040	40000040
i = 33	e0000052	e0000052
i = 34	a0000000	a0000000
i = 35	80000040	80000040
i = 36	20000001	20000001
i = 37	20000060	20000060
i = 38	80000001	80000001
i = 39	40000042	40000042

	Δm	Δm w/o carry
i = 40	c0000043	c0000043
i = 41	40000022	40000022
i = 42	00000003	00000003
i = 43	40000042	40000042
i = 44	c0000043	c0000043
i = 45	c0000022	c0000022
i = 46	00000001	00000001
i = 47	40000002	40000002
i = 48	c0000043	c0000043
i = 49	40000062	40000062
i = 50	80000001	80000001
i = 51	40000042	40000042
i = 52	40000042	40000042
i = 53	40000002	40000002
i = 54	00000002	00000002
i = 55	00000040	00000040
i = 56	80000002	80000002
i = 57	80000000	80000000
i = 58	80000002	80000002
i = 59	80000040	80000040
i = 60	00000000	00000000
i = 61	80000040	80000040
i = 62	80000000	80000000
i = 63	00000040	00000040
i = 64	80000000	80000000
i = 65	00000040	00000040
i = 66	80000002	80000002
i = 67	00000000	00000000
i = 68	80000000	80000000
i = 69	80000000	80000000
i = 70	00000000	00000000
i = 71	00000000	00000000
i = 72	00000000	00000000
i = 73	00000000	00000000
i = 74	00000000	00000000
i = 75	00000000	00000000
i = 76	00000000	00000000
i = 77	00000000	00000000
i = 78	00000000	00000000
i = 79	00000000	00000000

	Δa	Δa w/o carry
i = 0	00000000	00000000
i = 1	a8000041	a8000041
i = 2	80000803	80000801
i = 3	003f0012	00010012
i = 4	90200e00	90200200
i = 5	040fc00f	04004001
i = 6	02000010	02000010
i = 7	ffffffff	80000009
i = 8	00000002	00000002
i = 9	40000000	40000000
i = 10	00000000	00000000
i = 11	00000002	00000002
i = 12	80000001	80000001
i = 13	00000002	00000002
i = 14	00000000	00000000
i = 15	80000001	80000001
i = 16	00000000	00000000
i = 17	40000001	40000001
i = 18	00000002	00000002
i = 19	00000002	00000002
i = 20	80000002	80000002
i = 21	00000001	00000001
i = 22	00000000	00000000
i = 23	80000001	80000001
i = 24	00000002	00000002
i = 25	00000002	00000002
i = 26	00000002	00000002
i = 27	00000000	00000000
i = 28	00000000	00000000
i = 29	00000001	00000001
i = 30	00000000	00000000
i = 31	80000002	80000002
i = 32	00000002	00000002
i = 33	80000002	80000002
i = 34	00000000	00000000
i = 35	00000002	00000002
i = 36	00000000	00000000
i = 37	00000003	00000003
i = 38	00000000	00000000
i = 39	00000002	00000002

	Δa	Δa w/o carry
i = 40	00000002	00000002
i = 41	00000001	00000001
i = 42	00000000	00000000
i = 43	00000002	00000002
i = 44	00000002	00000002
i = 45	00000001	00000001
i = 46	00000000	00000000
i = 47	00000000	00000000
i = 48	00000002	00000002
i = 49	00000003	00000003
i = 50	00000000	00000000
i = 51	00000002	00000002
i = 52	00000002	00000002
i = 53	00000000	00000000
i = 54	00000000	00000000
i = 55	00000002	00000002
i = 56	00000000	00000000
i = 57	00000000	00000000
i = 58	00000000	00000000
i = 59	00000002	00000002
i = 60	00000000	00000000
i = 61	00000002	00000002
i = 62	00000000	00000000
i = 63	00000002	00000002
i = 64	00000000	00000000
i = 65	00000002	00000002
i = 66	00000000	00000000
i = 67	00000000	00000000
i = 68	00000000	00000000
i = 69	00000000	00000000
i = 70	00000000	00000000
i = 71	00000000	00000000
i = 72	00000000	00000000
i = 73	00000000	00000000
i = 74	00000000	00000000
i = 75	00000000	00000000
i = 76	00000000	00000000
i = 77	00000000	00000000
i = 78	00000000	00000000
i = 79	00000000	00000000
i = 80	00000000	00000000

How to find sufficient conditions on a_i?

- Ignore message expansion in this step

We will calculate sufficient conditions of chaining variables by adjusting b_i, c_i, d_i so that

$$\delta f(i, b_i, c_i, d_i) = \delta a_{i+1} - (\delta a_i \lll 5) - \delta e_i - \delta m_i.$$

In this calculation, we must adjust carry effect by hand, where we must take into account that when $\delta a_{i+1,j} = (\delta a_i \lll 5)_j = \delta e_{i,j} = \delta m_{i,j} = 0$, $\delta f(i, b_i, c_i, d_i)_j$ must be 0, not 1. Adjusting carry effect is difficult to calculate automatically.

Sufficient conditions of message *m* in 58-round SHA-1

message variable	31 - 24	23 - 16	15 - 8	8 - 0
m_0	--0-----	--------	--------	--------
m_1	-01-----	--------	--------	--01--1-
m_2	-10-----	--------	--------	-1----11
m_3	--0-----	--------	--------	-1------
m_4	000-----	--------	--------	-0----1-
m_5	-11-----	--------	--------	------1-
m_6	0-------	--------	--------	-------0
m_7	--------	--------	--------	--1-----
m_8	--------	--------	--------	------00
m_9	-0------	--------	--------	-0-1--1-
m_{10}	-0------	--------	--------	-0------
m_{11}	101-----	--------	--------	-1-1--1-
m_{12}	1-1-----	--------	--------	--------
m_{13}	0-------	--------	--------	-0------
m_{14}	--0-----	--------	--------	-------0
m_{15}	--0-----	--------	--------	-11-----
m_{16}	0-------	--------	--------	-------0
m_{17}	-0------	--------	--------	-1----0-
m_{18}	00------	--------	--------	-1----01
m_{19}	-0------	--------	--------	--1---1-
m_{20}	--------	--------	--------	------11
m_{21}	-0------	--------	--------	-0----1-
m_{22}	01------	--------	--------	-0----10

Sufficient conditions of chaining variables *a* in 58-round SHA-1

chaining variable	31 - 24	23 - 16	15 - 8	8 - 0
a_0	01100111	01000101	00100011	00000001
a_1	101-----	--------	--------	-1-a10aa
a_2	01100---	------0-	----a---	1--00010
a_3	0010----	-10---1a	------0-	0a-1a0-0
a_4	11010---	-01-----	01aaa---	0-10-100
a_5	10-01a--	-1-01-aa	--00100-	0---01-1
a_6	11--0110	-a-1001-	01100010	1-a111-1
a_7	-1--1110	a1a1111-	-101-001	1---0-10
a_8	-0----10	0000000a	a001a1--	100-0-1-
a_9	00------	11000100	00000000	101-1-1-
a_{10}	0-1-----	11111011	11100000	00--0-1-
a_{11}	1-0-----	-------1	01111110	11----0-
a_{12}	0-1-----	--------	--------	-1--a---
a_{13}	1-0-----	--------	--------	-1---01-
a_{14}	1-------	--------	--------	-1---1--
a_{15}	0-------	--------	--------	----0--0
a_{16}	-1------	--------	--------	----a---
a_{17}	-0------	--------	--------	----1-0-
a_{18}	1-1-----	--------	--------	----a-0-
a_{19}	--------	--------	--------	-------0
a_{20}	-C------	--------	--------	----A---
a_{21}	--------	--------	--------	----a-1-

‘a’: $a_{i,j} = a_{i-1,j}$

‘A’: $a_{i,j} = a_{i-1,j} + 1$

‘b’: $a_{i,j} = a_{i-1,(j+2) \bmod 32}$

‘B’: $a_{i,j} = a_{i-1,(j+2) \bmod 32} + 1$

‘c’: $a_{i,j} = a_{i-2,(j+2) \bmod 32}$

‘C’: $a_{i,j} = a_{i-2,(j+2) \bmod 32} + 1$

Procedures for Message modification

- Our method
  - Gröbner Basis Based Method

Two Elimination Orders

- Elimination order of *m*

Here we introduce elimination order of $\{m_{i,j}\}\{i = 0, 1, \cdots, 15, j = 0, 1, \cdots, 31\}$ by

$$m'_{i',j'} \leq m_{i,j} \text{ if } i' \leq i \text{ or } (i' = i \text{ and } j' \leq j).$$

- Elimination order of *a*

Similarly we can consider different elimination order of $a_{i,j}\{i = 0, 1, \cdots, 15, j = 0, 1, \cdots, 31\}$ by

$$a'_{i',j'} \leq a_{i,j} \text{ if } i' \leq i \text{ or } (i' = i \text{ and } j' \leq j).$$

These two orders are different but approximately similar because transformation between them is not so complicated.

Two message modification techniques

- Modification of *a*
  - Decode as codes defined on *a*
- Modification of *m*
  - Decode as codes defined on *m*
- We use modification of *a*

Relations in 0-15-round of *m*

- All conditions on 0-57-round of *m* can be rewritten by 0-15-round relations
  - Using the relations derived of key expansion

    $$m_i = (m_{i\text{-}3} \oplus m_{i\text{-}8} \oplus m_{i\text{-}14} \oplus m_{i\text{-}16}) \lll 1$$

  - Using Gaussian elimination
  - Introduce elimination order of $\{m_{i,j}\}$ $\{i = 0,1,\ldots,15,\ j = 0,1,\ldots,31\}$ by

    $m'_{i',j'} \leq m'_{i',j'}$ if $i' \leq i$ or $(i' = i$ and $j' \leq j)$

Relation of 0-15-round of *m*

$$m_{15,31} = 1, m_{15,30} = 1, m_{15,29} = 0, m_{15,28} + m_{10,28} + m_{8,29} + m_{7,29} +$$
$$m_{4,28} + m_{2,28} = 1, m_{15,27} + m_{14,25} + m_{12,28} + m_{12,26} + m_{10,28} + m_{9,27} +$$
$$m_{9,25} + m_{8,29} + m_{8,28} + m_{7,28} + m_{7,27} + m_{6,26} + m_{5,28} + m_{4,26} + m_{3,25} +$$
$$m_{2,28} + m_{1,25} + m_{0,28} = 1, m_{15,26} + m_{10,28} + m_{10,26} + m_{8,28} + m_{8,27} +$$
$$m_{7,27} + m_{6,29} + m_{5,27} + m_{4,26} + m_{2,27} + m_{2,26} + m_{0,27} = 1, m_{15,25} +$$
$$m_{11,28} + m_{10,27} + m_{10,25} + m_{9,28} + m_{8,27} + m_{8,26} + m_{7,26} + m_{6,29} + m_{6,28} +$$
$$m_{5,26} + m_{4,25} + m_{3,28} + m_{2,28} + m_{2,26} + m_{2,25} + m_{1,28} + m_{0,28} + m_{0,26} =$$
$$0, m_{15,24} + m_{12,28} + m_{11,27} + m_{10,26} + m_{10,24} + m_{9,28} + m_{9,27} + m_{8,29} +$$
$$m_{8,26} + m_{8,25} + m_{7,25} + m_{6,29} + m_{6,28} + m_{6,27} + m_{5,25} + m_{4,28} + m_{4,24} +$$
$$m_{3,28} + m_{3,27} + m_{2,27} + m_{2,25} + m_{2,24} + m_{1,28} + m_{1,27} + m_{0,27} + m_{0,25} =$$
$$1, m_{15,23} + m_{12,28} + m_{12,27} + m_{11,26} + m_{10,25} + m_{10,23} + m_{9,27} + m_{9,26} +$$
$$m_{8,28} + m_{8,25} + m_{8,24} + m_{7,29} + m_{7,24} + m_{6,28} + m_{6,27} + m_{6,26} + m_{5,24} +$$
$$m_{4,27} + m_{4,23} + m_{3,27} + m_{3,26} + m_{2,26} + m_{2,24} + m_{2,23} + m_{1,27} + m_{1,26} +$$
$$m_{0,26} + m_{0,24} = 1, m_{15,22} + m_{14,25} + m_{12,28} + m_{12,27} + m_{11,25} + m_{10,27} +$$
$$m_{10,24} + m_{10,22} + m_{9,28} + m_{9,27} + m_{9,26} + m_{8,27} + m_{8,24} + m_{8,23} +$$
$$m_{7,28} + m_{7,27} + m_{7,23} + m_{6,27} + m_{6,25} + m_{5,23} + m_{4,28} + m_{4,27} + m_{4,22} +$$
$$m_{3,26} + m_{2,28} + m_{2,27} + m_{2,25} + m_{2,23} + m_{2,22} + m_{1,26} + m_{0,25} + m_{0,23} =$$
$$0, m_{15,6} = 1, m_{15,5} = 1, m_{15,4} + m_{12,5} + m_{10,4} + m_{4,5} + m_{4,4} + m_{2,5} + m_{2,4} =$$

Control sequence (I)

Control sequence s_i	Control bit b_i	Controlled relation r_i
s_{120}	$a_{16,31}$	$m_{15,31} = 1$
s_{119}	$a_{16,29}$	$m_{15,29} = 0$
s_{118}	$a_{16,28}$	$m_{15,28} + m_{10,28} + m_{8,29} + m_{7,29} + m_{4,28} + m_{2,28} = 1$
s_{117}	$a_{16,27}$	$m_{15,27} + m_{14,25} + m_{12,28} + m_{12,26} + m_{10,28} + m_{9,27} + m_{9,25} + m_{8,29} + m_{8,28} + m_{7,28} + m_{7,27} + m_{6,26} + m_{5,28} + m_{4,26} + m_{3,25} + m_{2,28} + m_{1,25} + m_{0,28} = 1$
s_{116}	$a_{16,26}$	$m_{15,26} + m_{10,28} + m_{10,26} + m_{8,28} + m_{8,27} + m_{7,27} + m_{6,29} + m_{5,27} + m_{4,26} + m_{2,27} + m_{2,26} + m_{0,27} = 1$
s_{115}	$a_{16,25}$	$m_{15,25} + m_{11,28} + m_{10,27} + m_{10,25} + m_{9,28} + m_{8,27} + m_{8,26} + m_{7,26} + m_{6,29} + m_{6,28} + m_{5,26} + m_{4,25} + m_{3,28} + m_{2,28} + m_{2,26} + m_{2,25} + m_{1,28} + m_{0,28} + m_{0,26} = 0$
s_{114}	$a_{16,24}$	$m_{15,24} + m_{12,28} + m_{11,27} + m_{10,26} + m_{10,24} + m_{9,28} + m_{9,27} + m_{8,29} + m_{8,26} + m_{8,25} + m_{7,25} + m_{6,29} + m_{6,28} + m_{6,27} + m_{5,25} + m_{4,28} + m_{4,24} + m_{3,28} + m_{3,27} + m_{2,27} + m_{2,25} + m_{2,24} + m_{1,28} + m_{1,27} + m_{0,27} + m_{0,25} = 1$
s_{113}	$a_{16,23}$	$m_{15,23} + m_{12,28} + m_{12,27} + m_{11,26} + m_{10,25} + m_{10,23} + m_{9,27} + m_{9,26} + m_{8,28} + m_{8,25} + m_{8,24} + m_{7,29} + m_{7,24} + m_{6,28} + m_{6,27} + m_{6,26} + m_{5,24} + m_{4,27} + m_{4,23} + m_{3,27} + m_{3,26} + m_{2,26} + m_{2,24} + m_{2,23} + m_{1,27} + m_{1,26} + m_{0,26} + m_{0,24} = 1$
s_{112}	$a_{16,22}$	$m_{15,22} + m_{14,25} + m_{12,28} + m_{12,27} + m_{11,25} + m_{10,27} + m_{10,24} + m_{10,22} + m_{9,28} + m_{9,27} + m_{9,26} + m_{8,27} + m_{8,24} + m_{8,23} + m_{7,28} + m_{7,27} + m_{7,23} + m_{6,27} + m_{6,25} + m_{5,23} + m_{4,28} + m_{4,27} + m_{4,22} + m_{3,26} + m_{2,28} + m_{2,27} + m_{2,25} + m_{2,23} + m_{2,22} + m_{1,26} + m_{0,25} + m_{0,23} = 0$
s_{111}	$a_{16,21}$	$a_{18,31} = 1$
[illegible]	[illegible]	[illegible]

Control Sequence (II)

Control sequence s_i	Control bit b_i	Controlled relation r_i
s_{82}	$a_{14,30}$	$m_{14,3} + m_{11,3} + m_{11,2}$ $+m_{8,2} + m_{7,4} + m_{7,2} + m_{7,1} + m_{6,2} + m_{5,3}$ $+m_{4,0} + m_{3,3} + m_{2,2} + m_{1,31} + m_{1,3} = 0$
s_{81}	$a_{15,2}$	$m_{14,2} + m_{12,5} + m_{12,3} + m_{10,4} + m_{9,2} + m_{7,4}$ $+m_{6,3} + m_{4,5} + m_{4,4} + m_{4,3} + m_{3,2} + m_{2,5}$ $+m_{2,4} + m_{1,2} = 1$
s_{80}	$a_{15,1}$	$m_{14,1} + m_{12,4} + m_{11,2} + m_{10,2} + m_{9,3} + m_{8,3}$ $+m_{7,2} + m_{6,2} + m_{5,5} + m_{5,2} + m_{4,4} + m_{3,31}$ $+m_{3,4} + m_{3,2} + m_{3,1} + m_{2,4} + m_{2,3} + m_{0,3} = 0$
s_{79}	$a_{14,27}$	$m_{14,0} = 0$
s_{78}	$a_{13,26}$	$m_{13,31} = 0$
s_{77}	$a_{13,25}$	$m_{13,30} = 0$
s_{76}	$a_{14,29}$	$m_{13,29} + m_{8,29} = 0$
s_{75}	$a_{14,28}$	$m_{13,28} + m_{8,28} + m_{2,28} + m_{0,28} = 0$
s_{74}	$a_{13,22}$	$m_{13,27} + m_{11,28} + m_{8,29} + m_{8,27} + m_{6,29}$ $+m_{5,28} + m_{3,28} + m_{2,27} + m_{0,27} = 1$
s_{73}	$a_{13,21}$	$m_{13,26} + m_{11,27} + m_{9,28} + m_{8,28} + m_{8,26}$ $+m_{6,28} + m_{5,27} + m_{3,28} + m_{3,27} + m_{2,26}$ $+m_{1,28} + m_{0,26} = 1$
s_{72}	$a_{14,24}$	$m_{13,24} + m_{12,28} + m_{11,27} + m_{11,25} + m_{10,28}$ $+m_{9,27} + m_{9,26} + m_{8,29} + m_{8,26} + m_{8,24}$ $+m_{7,29} + m_{7,28} + m_{6,26} + m_{5,25} + m_{4,28}$ $+m_{3,28} + m_{3,26} + m_{3,25} + m_{2,28} + m_{2,24}$ $+m_{1,28} + m_{1,26} + m_{0,24} = 0$
s_{71}	$a_{14,23}$	$m_{13,23} + m_{12,27} + m_{11,26} + m_{11,24} + m_{10,28}$ $+m_{10,27} + m_{9,26} + m_{9,25} + m_{8,29} + m_{8,28}$ $+m_{8,25} + m_{8,23} + m_{7,29} + m_{7,28} + m_{7,27}$ $+m_{6,25} + m_{5,28} + m_{5,24} + m_{4,28} + m_{4,27}$

Control Sequence (III)

Control sequence s_i	Control bit b_i	Controlled relation r_i
s_{22}	$a_{5,25}$	$m_{5,30} = 1$
s_{21}	$a_{6,29}$	$m_{5,29} = 1$
s_{20}	$a_{6,1}$	$m_{5,1} = 1$
s_{19}	$a_{3,27}$	$m_{5,0} + m_{3,0} + m_{1,31} = 1$
s_{18}	$a_{4,26}$	$m_{4,31} = 0$
s_{17}	$a_{4,25}$	$m_{4,30} = 0$
s_{16}	$a_{5,29}$	$m_{4,29} = 0$
s_{15}	$a_{5,6}$	$m_{4,6} = 0$
s_{14}	$a_{5,1}$	$m_{4,1} = 1$
s_{13}	$a_{3,25}$	$m_{3,30} = 1$
s_{12}	$a_{3,24}$	$m_{3,29} = 0$
s_{11}	$a_{4,6}$	$m_{3,6} = 1$
s_{10}	$a_{2,26}$	$m_{2,31} = 0$
s_9	$a_{2,25}$	$m_{2,30} = 1$
s_8	$a_{2,24}$	$m_{2,29} = 0$
s_7	$a_{3,5}$	$m_{2,6} = 1$
s_6	$a_{2,6}$	$m_{2,6} = 1$
s_5	$a_{3,1}$	$m_{2,1} = 1$
s_4	$a_{2,5}$	$m_{1,5} = 0$
s_3	$a_{1,28}$	$m_{1,1} = 1$
s_2	$a_{1,25}$	$m_{1,30} = 0$
s_1	$a_{1,24}$	$m_{1,29} = 1$
s_0	$a_{1,23}$	$m_{1,29} = 1$

Table 6 Control bit and controlled relations of 58-round SHA-1 (III)

Advanced sufficient conditions of message *m*

message variable	31 - 24	23 - 16	15 - 8	8 - 0
m_0	--0-----	--------	--------	--------
m_1	-01-----	--------	--------	--01--1-
m_2	L10-----	--------	--------	-1----11
m_3	-L0-----	--------	--------	-1------
m_4	000-----	--------	--------	-0----1-
m_5	L11-----	--------	--------	------1L
m_6	0L------	--------	--------	-------0
m_7	LL------	--------	--------	--1----L
m_8	LL------	--------	--------	------00
m_9	L0L-----	--------	--------	-0L1--1L
m_{10}	L0L-----	--------	--------	-0L----L
m_{11}	101-----	--------	--------	-1-1--1L
m_{12}	1L1-----	--------	--------	-------L
m_{13}	0LLLLL-L	LL------	--------	-0LLLLLL
m_{14}	LL0LLL-L	LLLL----	--------	--LLLLL0
m_{15}	LL0LLLLL	LL------	--------	-11LLLLL
m_{16}	0-------	--------	--------	-------0
m_{17}	-0------	--------	--------	-1----0-
m_{18}	00------	--------	--------	-1----01
m_{19}	-0------	--------	--------	--1---1-
m_{20}	--------	--------	--------	------11
m_{21}	-0------	--------	--------	-0----1-
m_{22}	01------	--------	--------	-0----10

Advanced sufficient conditions of chaining variable *a*

chaining variable	31 - 24	23 - 16	15 - 8	8 - 0
a_0	01100111	01000101	00100011	00000001
a_1	101V--vV	Y-------	--------	-1-a10aa
a_2	01100vVv	------0-	----a---	1-w00010
a_3	0010--Vv	-10---1a	------0-	0aX1a0W0
a_4	11010vv-	-01-----	01aaa---	0W10-100
a_5	10w01aV-	-1-01-aa	--00100-	0w--01W1
a_6	11W-0110	-a-1001-	01100010	1-a111W1
a_7	w1x-1110	a1a1111-	-101-001	1---0-10
a_8	h0Xvvv10	0000000a	a001a1--	100X0-1h
a_9	00XVrrvV	11000100	00000000	101-1-1y
a_{10}	0w1-rv-v	11111011	11100000	00hW0-1r
a_{11}	1w0--V-V	-------1	01111110	11x---0Y
a_{12}	0w1-rV-V	--------	--------	-1XWa-Wh
a_{13}	1w0--vv-	-rr-----	--------	-1---01y
a_{14}	1rhhvvVh	hh------	--------	-1hhh1hh
a_{15}	0rwhhhVh	hhhh----	--------	--hh0hh0
a_{16}	W1whhhhh	hhq-q-q-	q--q-qqq	-WWhahhh
a_{17}	-0------	--------	--------	----1-0-
a_{18}	1-1-----	--------	--------	------0-
a_{19}	--------	--------	--------	-------0
a_{20}	--------	--------	--------	--------
a_{21}	--------	--------	--------	------1-

1, 0, a: Wang's sufficient conditions

w: adjust $a_{i+1,j}$ so as $m_{i,j} = 0$

W: adjust $a_{i+1,j}$ so as $m_{i,j} = 1$

v: adjust $a_{i,j-5}$ so as $m_{i,j} = 0$

V: adjust $a_{i,j-5}$ so as $m_{i,j} = 1$

'h': adjust *ai*,*j* so that corresponding controlled relation including $m_{i+1,j}$ as leading term holds

'r': adjust $a_{i,j}$ so that corresponding controlled relation including $m_{i,(j+27)\mathrm{mod}32}$ as leading term holds

…

Improvement of Message Modification technique

- Success probability is not 1
  - Control sequences sometimes rotate and do not end
  - Changing control bits may not affect leading term properly
- New method
  - Multiple control bits
    - Use iterative decoding technique
    - Use list decoding technique
  - Controlling non-leading terms
  - Using semi-neutral bits

Neutral bit

- Introduced by Biham and Chen
- Some bits do not affect relations
  - Increase the probability of collision

Semi-neutral bit

- We introduce new notion ‘Semi-neutral bit’
- Change of some bits can easily be adjusted in a few steps of control sequence
  - Which means that noise on semi-neutral bits can be easily corrected

Sufficient conditions and new message modification techniques

chaining variable	31 - 24	23 - 16	15 - 8	8 - 0
a_0	01100111	01000101	00100011	00000001
a_1	101V--vV	Y-------	--------	-1-a10aa
a_2	01100vVv	------0-	----a---	1-w00010
a_3	0010--Vv	-10---1a	------0-	0aX1a0W0
a_4	11010vv-	-01-----	01aaa---	0W10-100
a_5	10w01aV-	-1-01-aa	--00100-	0w--01W1
a_6	11W-0110	-a-1001-	01100010	1-a111W1
a_7	w1x-1110	a1a1111-	-101-001	1---0-10
a_8	h0Xvvv10	0000000a	a001a1--	100X0-1h
a_9	00XVrr-V	11000100	00000000	101-1-1y
a_{10}	0w1-rv-v	11111011	11100000	00hW0-1h
a_{11}	1w0--V-V	-------1	01111110	11x---0Y
a_{12}	0w1-rV-V	--------	--------	-1XWa-Wh
a_{13}	1w0--vv-	-rr-----	--------	-1-qq01y
a_{14}	1rhhvvVh	hh------	qNNNNNqN	N1hhh1hh
a_{15}	0rwhhhVh	hhhh---N	qNNqqNqN	NNhh0hh0
a_{16}	W1whhhhh	hhqNqNqN	NNqNNqqq	qWWhahhh
a_{17}	-0------	--------	--------	----100-
a_{18}	1-1-----	--------	--------	-----00-
a_{19}	--------	--------	--------	-------0

1, 0, a: Wang's sufficient conditions

w: adjust $a_{i+1,j}$ so that $m_{i,j} = 0$

W: adjust $a_{i+1,j}$ so that $m_{i,j} = 1$

v: adjust $a_{i,j-5}$ so that $m_{i,j} = 0$

V: adjust $a_{i,j-5}$ so that $m_{i,j} = 1$

N: semi-neutral bit

…

We propose the method to determine sufficient conditions and new message modification technique using Gröbner basis

New collision example of 58-step SHA-1

M = 0x

1ead6636 319fe59e 4ea7ddcb c7961642 0ad9523a f98f28db 0ad135d0 e4d62aec 6c2da52c 3c7160b6 06ec74b2 b02d545e bdd9e466 3f156319 4f497592 dd1506f93

M' = 0x

ead6636 519fe5ac 2ea7dd88 e7961602 ead95278 998f28d9 8ad135d1 e4d62acc 6c2da52f 7c7160e4 46ec74f2 502d540c 1dd9e466 bf156359 6f497593 fd150699

- Note that the proposed method is the first fully-published method that can cryptanalyze 58-round SHA-1

Further improvement: Using Groebner base based method (Algorithm 3)

chaining variable	31 - 24	23 - 16	15 - 8	8 - 0
a_0	01100111	01000101	00100011	00000001
a_1	101V--vV	Y-------	--------	-1-a10aa
a_2	01100vVv	------0-	----a---	1-w00010
a_3	0010--Vv	-10---1a	------0-	0aX1a0W0
a_4	11010vv-	-01-----	01aaa---	0W10-100
a_5	10w01aV-	-1-01-aa	--00100-	0w--01W1
a_6	11W-0110	-a-1001-	01100010	1-a111W1
a_7	w1x-1110	a1a1111-	-101-001	1---0-10
a_8	h0Xvvv10	0000000a	a001a1--	100X0-1h
a_9	00XVrr-V	11000100	00000000	101-1-1y
a_{10}	0w1-rv-v	11111011	11100000	00hW0-1h
a_{11}	1w0--V-V	-------1	01111110	11x---0Y
a_{12}	0w1-rV-V	--------	--------	-1XWa-Wh
a_{13}	1w0--vv-	-rr-----	--------	-1-qq01y
a_{14}	1rhhvvVh	hh------	qNNNNNqN	N1hhh1hh
a_{15}	0rwhhhVh	hhhh---N	qNNqqNqN	NNhh0hh0
a_{16}	W1whhhhh	hhqNqNqN	NNqNNqqq	qWWhahhh
a_{17}	-0------	--------	--------	----100-
a_{18}	1-1-----	--------	--------	-----00-
a_{19}	--------	--------	--------	-------0

Problem to determine semi-neutral bits denoted as ‘N’ is equivalent to calculating Groebner basis from algebraic equations on variable denoted as ‘q’ or ‘N’

Calculation of Groebner basis

Cryptanalysis of 58-round SHA-1

- We can achieve all message conditions and 8 chaining value conditions in 17 – 23 round (success probability is 0.5)
- 29 conditions remained
  - > exhaustive search (2^{29} message modification)
  - Constant is practical
- Utilization of Groebner base based method
- 2^{29} message modification -> 2^{8} message modification (symbolic computation)
- However, complexity is exactly same
  - 2^{29} SHA-1 -> 2^{29} SHA-1
- Complexity can be reduced employing a suitable technique of error correcting code and Groebner basis

Cryptanalysis of full-round SHA-1 (first iteration)

- We can achieve all message conditions and all chaining variable conditions in 17 – 26 round
- 64 conditions remained
  - > exhaustive search (2^{64} message modification)
  - Constant is practical?
- Utilization of Groebner base based method
- 2^{64} message modification -> 2^{51} message modification (symbolic computation)
- However, total complexity is still same
- Complexity can be reduced employing a suitable technique of error correcting code and Groebner basis?

Example which satisfies sufficient conditions until 28-th round

M = 0x

aa740c82 9f91e819 84c3e50f a898306b
1e5b4111 1867d96b 0616ea95 014a2f32
7ae92980 d5e4d6c6 9d49d0ba 3b8087d3
32717277 edcec899 dc537498 63bca615

- The above M satisfies all message conditions of 0-80 rounds and all chaining variable conditions of 0-28 rounds

Conclusion

- Proposed the novel method for finding the differential characteristic, method for determining sufficient conditions and the novel method for the message modification using Gröbner-like method
- Succeeded in finding collisions of 58-step SHA-1
  - Showed by experiments the efficiency of proposed method