Pushing Collision Attacks on SHA-2 to 39 Steps

Yingxin Li[1], Zhuolong Zhang[2], Muzhou Li[3], Fukang Liu[4], Haifeng Qian[1] and Jinwei Zhu[1]

[1] School of Cryptology, East China Normal University, Shanghai, China
yxli@sc.ecnu.edu.cn,
[2] School of Cyber Science and Technology, Shandong University, Qingdao, China
zhuolongzhang@mail.sdu.edu.cn
[3] School of Cryptologic Science and Engineering, Shandong University, Jinan, China
muzhouli@mail.sdu.edu.cn
[4] Institute of Science Tokyo, Tokyo, Japan
liu.f.ad@m.titech.ac.jp

Abstract. The SHA-2 family is a U.S. federal standard and mainly includes SHA-256 and SHA-512. In particular, SHA-256 plays a central role in real-world applications and is widely regarded as one of the most important hash functions in use today. At CRYPTO 2026, Li et al. proposed collision attacks up to 37-step SHA-2, but they could not reach 38 steps due to the low-probability uncontrolled part in the corresponding differential characteristics. In this paper, we propose an improved search procedure to find high-quality differential characteristics for 38-step SHA-256 and SHA-512, respectively. Exploiting the special shape of the 38-step differential characteristics, the meet-in-the-middle method to fulfill the corresponding differential conditions is extremely memory-efficient. Consequently, we successfully achieve the first 38-step collision attack on both SHA-256 and SHA-512, whose time complexity is $2^{104.3}$ and $2^{125.4}$, respectively. The memory complexity of the 38-step collision attack is negligible. The methods are also applied to the 36-step and 37-step collision attacks on SHA-2 published at CRYPTO 2026, leading to a significant improvement in both time and memory complexity. In particular, the time complexity of the collision attack on 36-step SHA-256 is only 2^{57} and the memory complexity is negligible. We have verified this collision attack by providing the first colliding message pair for 36-step SHA-256. More remarkably, we apply the new method to 39 steps of SHA-2 and obtain the first effective collision attack on 39-step SHA-512, with a time complexity of 2^{178} and negligible memory complexity. However, the method does not yield an effective collision attack on 39-step SHA-256 due to the same issue arising from the low-probability uncontrolled part. Overall, this work further pushed the limit of memory-efficient collision attacks on round-reduced SHA-2 and significantly advances the state of the art.

Keywords: hash function · SHA-2 · meet-in-the-middle · collision attack

1 Introduction

The MD-SHA family of hash functions has long been an important foundation of modern cryptography. However, since Wang *et al.* proposed new cryptanalytic techniques such as signed differences and message modification [WLF+05, WYY05a, WY05, WYY05b], the security of the MD-SHA family has been severely impacted. These methods have greatly advanced the development of collision attacks on hash functions and led to the successive breaking of several algorithms, including MD4 [WLF+05], MD5 [WY05], RIPEMD [WLF+05], SHA-0 [WYY05b], SHA-1 [WYY05a, SBK+17], and RIPEMD-128 [MNS12]. In contrast,

RIPEMD-160 and the SHA-2 family have withstood the attacks, and their full versions still maintain a substantial security margin against collision attacks. As SHA-2 is widely deployed in critical scenarios such as digital signatures, secure communication protocols, software integrity verification, and blockchain systems, and in particular as SHA-256 is a core cryptographic component in Bitcoin. Therefore, the security of SHA-2 has continuously attracted extensive attention from both academia and industry.

Traditionally, such a cryptographic hash function has to fulfill the three basic requirements of collision resistance, preimage resistance and second preimage resistance in order to be considered secure. Among them, collision resistance requires that it is difficult for an attacker to find two distinct messages that produce the same hash value. Since SHA-2 adopts a complex message expansion and an iterative compression structure, its internal state update involves various nonlinear and linear operations, including modular addition, XOR, word-wise rotations, and Boolean functions. This makes collision attacks on SHA-2 highly challenging. For a long time, how to construct high-probability differential characteristics, how to effectively control differential propagation in the message expansion, and how to reduce the time and space complexity of message modification have been core problems in the cryptanalysis of SHA-2.

In the study of collision attacks on SHA-2, before the year 2008, the number of attacked steps reached only 24 steps for SHA-256 and SHA-512 [IMPR08, SS08]. Since the year 2011, collision attacks on SHA-2 have often relied on advanced automatic search tools and novel constructions of local collisions. Specifically, at ASIACRYPT 2011, the first major improvement was achieved, where the advanced guess-and-determine (GnD) technique was invented to search differential characteristics for SHA-256 [MNS11], where a semi-free-start (SFS) collision for 32-step SHA-256 and a collision for 27-step SHA-256 were presented, respectively. With the help of this tool, several key advances were made in collision attacks on SHA-2. Specifically, at EUROCRYPT 2013, Mendel *et al.* [MNS13] proposed new constructions of local collisions and achieved a practical collision attack on 28-step SHA-256 and a theoretical collision attack on 31-step SHA-256. At FSE 2014, Eichlseder *et al.* [EMS14] found a practical SFS collision attack on 38-step SHA-512. At ASIACRYPT 2015, Dobraunig *et al.* [DEM15] further achieved a practical SFS collision attack on 39-step SHA-512 and a practical collision attack on 27-step SHA-512.

However, the above advanced automatic tool for SHA-2 was not made public until 2024[1]. In another line of research to develop automatic tools for the MD-SHA hash family, Liu *et al.* first invented a novel MILP-based technique to search for signed differential characteristics of RIPEMD-160 at EUROCRYPT 2023 [LWS+23]. Then, Li *et al.* reimplemented this method using SAT/SMT and applied it to both RIPEMD-160 [LLW23] and SHA-2 [LLW24a]. With this SAT/SMT-based tool, they achieved the first practical SFS collision attack on 39-step SHA-256, an improved theoretical collision attack on 31-step SHA-512, and a practical free-start (FS) collision attack on 40-step SHA-256 [LLW24a]. In addition, to address the issue of vast memory consumption in Mendel et al.'s meet-in-the-middle technique to fulfill differential conditions [MNS13], Li *et al.* [LLW+24b] proposed a new memory-efficient collision attack framework for SHA-2, leading to the first practical collision attack on 31-step SHA-256 [LLW+24b].

The MILP/SAT/SMT-based tools [LWS+23, LLW23, LLW24a] for the MD-SHA hash family and the memory-efficient framework for SHA-2 [LLW+24b] have led to significant progress in collision attacks on SHA-2. At CRYPTO 2025, the first practical collision attack on 29-step SHA-512 was achieved [LLW+25]. At EUROCRYPT 2026, Zhang *et al.* [ZMLM26] further revised the search model in [LLW+25], and achieved the first theoretical collision attacks on 36/37-step SHA-256. At CRYPTO 2026, Li *et al.* [LLWS26] investigated various constructions of local collisions for SHA-2 in the literature, and proposed the first practical collision attacks on 35-step SHA-2, as well as theoretical

[1] https://github.com/iaikkrypto/nldtool

collision attacks on 36/37-step SHA-2. Although the method in [LLWS26] naturally extends to the 38 steps, the bottleneck becomes the probability of the uncontrolled part in the corresponding 38-step differential characteristics. Although the authors provided the differential characteristics for 38-step SHA-256 and SHA-512 in [LLWS26], the time complexity of the 38-step collision attack based on them exceeds the generic birthday attack bound. Therefore, how to find improved 38-step differential characteristics becomes the key to achieving the first collision attack on 38-step SHA-2.

Our contributions. The contributions of this paper are summarized as below and the results can be found in Table 1.

1. Following the method in [LLWS26] to inject differences in message words

$$(W_7, \ldots, W_{11}, W_{15}, W_{16}, W_{23}, W_{25}),$$

   we propose a more refined search procedure to search for 38-step differential characteristics. Based on an improved memory-efficient collision attack framework, we finally achieve the first theoretical collision attack on 38-step SHA-2. The time and memory complexity can be found in Table 1.

2. The refined search procedure is generalized and applied to finding improved differential characteristics for 36/37-step SHA-2. As a result, the time and memory complexity of the 36/37-step collision attacks are further reduced. In particular, we achieved this first practical collision attack on 36-step SHA-256, as shown in Table 2.

3. The new general search procedure is further applied to 39-step SHA-2. Although an effective collision attack on 39-step SHA-256 cannot be achieved due to the low probability of the uncontrolled part, we obtain the first effective collision attack on 39-step SHA-512. This result represents a significant milestone in the line of research on MILP/SAT/SMT-based techniques for finding signed differential characteristics [LWS+23, LLW24a, LLW+24b, LLW+25, ZMLM26], as collision attacks on SHA-512 now reach the same number of steps as the best SFS collision attack that was published at ASIACRYPT 2015 [DEM15].

Organization. We first briefly describe the specification of SHA-256 and SHA-512 in Section 2. Then, we present a new search strategy to find 38-step differential characteristics in Section 3. This search strategy is generalized in Section 4 and applied to 36/37/39-step SHA-2. Finally, the paper is concluded in Section 5. For convenience, all tables for the obtained differential characteristics, their SFS colliding message pairs and extra conditions affecting the time complexity of the collision attacks are placed in Appedix A.

2 Preliminaries

2.1 Notations

For a better understanding of this paper, we introduce the following notations.

1. $\boxplus$ and $\boxminus$ represent modulo addition and modulo subtraction on 32/64 bits, respectively.
2. $\gg$, $\ggg$, $\oplus$, $\neg$, $\vee$ and $\wedge$ represent *shift right*, *rotate right*, *exclusive or*, *not*, *or*, and *and*, respectively.
3. $x[i]$ denotes the i-th bit of x and $x[0]$ is the least significant bit.
4. δx denotes the modular difference, i.e., $\delta x = x' \boxminus x$.

Table 1: Summary of (semi-free-start) collision attacks on SHA-2.

Version	Attack type	Steps	Time	Memory	References
SHA-256	collision	28	*practical*		[MNS13]
		29	*practical*		[LLW+25]
		31	$2^{65.5}$	2^{34}	[MNS13]
		31	$2^{49.8}$	2^{48}	[LLW24a]
		31	*practical*		[LLW+24b]
		35	*practical*		[LLWS26]
		36	$2^{94.4}$	$2^{12.6}$	[ZMLM26]
		36	2^{71}	$2^{38.585}$	[LLWS26]
		36	***practical***		**Sect. 4**
		37	$2^{122.2}$	$2^{34.9}$	[ZMLM26]
		37	$2^{119.1}$	$2^{39.9}$	[ZMLM26]
		37	2^{81}	$2^{53.047}$	[LLWS26]
		37	$\mathbf{2^{79.1}}$	negligible	**Sect. 4**
		38	$\mathbf{2^{104.3}}$	negligible	**Sect. 3**
	SFS collision	38	*practical*		[MNS13]
		39	*practical*		[LLW24a]
SHA-512	collision	27	*practical*		[DEM15]
		28	*practical*		[LLW24a]
		29	*practical*		[LLW+25]
		31	$2^{115.6}$	$2^{77.3}$	[LLW24a]
		31	$2^{97.3}$	$2^{35.2}$	[LLW+24b]
		31	$2^{85.5}$	$2^{44.4}$	[LLW+25]
		35	*practical*		[LLWS26]
		36	2^{136}	2^{87}	[LLWS26]
		36	$\mathbf{2^{66.2}}$	negligible	**Sect. 4**
		37	2^{185}	2^{132}	[LLWS26]
		37	$\mathbf{2^{87}}$	negligible	**Sect. 4**
		38	$\mathbf{2^{125.4}}$	negligible	**Sect. 3**
		39	$\mathbf{2^{178}}$	negligible	**Sect. 4**
	SFS collision	38	*practical*		[EMS14]
		39	*practical*		[DEM15]

5. ∇x denotes the signed difference between x' and x. We use the same notation as in [LMS+22, LWS+23], i.e.,

$$\nabla x[i] = \begin{cases} \texttt{n} & (x[i] = 0, x'[i] = 1) \\ \texttt{u} & (x[i] = 1, x'[i] = 0) \\ \texttt{=} & (x[i] = x'[i]) \\ \texttt{0} & (x[i] = x'[i] = 0) \\ \texttt{1} & (x[i] = x'[i] = 1) \end{cases} \tag{1}$$

6. $M = (m_0, m_1, \ldots, m_{15})$ and $M' = (m'_0, m'_1, \ldots, m'_{15})$ represent two message blocks.

Definition 1. [LWS+23] The signed difference ∇x is said to be an expansion of the modular difference δx only when ∇x corresponds to the modular difference δx.

Definition 2. [LWS+23] The hamming weight of the signed difference ∇x is denoted by $\mathbf{H}(\nabla x)$ and $\mathbf{H}(\nabla x)$ is the number of indices i such that $\nabla x[i] \in \{\texttt{n}, \texttt{u}\}$.

For example, let

$$\nabla x_0 = [\texttt{==== nu== ==== ==== ==== ==== ==== ====}],$$
$$\nabla x_1 = [\texttt{==== =n== ==== ==== ==== ==== ==== ====}].$$

Then, both ∇x_0 and ∇x_1 are the expansions of $\delta x = 2^{26}$. Moreover, we have $\mathbf{H}(\nabla x_0) = 2$ and $\mathbf{H}(\nabla x_1) = 1$. As each signed difference corresponds to a unique modular difference, for convenience, when computing $\delta x \boxplus \delta y$ for a given $(\nabla x, \nabla y)$, we also simply denote $\delta x \boxplus \delta y$ by $\nabla x \boxplus \nabla y$. For the above example, we have $\nabla x_0 \boxplus \nabla x_1 = 2^{27}$.

2.2 Description of SHA-2

The SHA-2 family is a series of hash functions standardized by NIST as part of the Secure Hash Standard (SHS) [Nat12]. SHA-2 adopts the well-known Merkle-Damgård construction [Dam89, Mer89], and its compression functions employ the Davies-Meyer construction. We use n to represent the state words of SHA-2, where $n = 32$ for SHA-256 and $n = 64$ for SHA-512. SHA-2 is an iterated hash function that pads and processes the input message using t $16n$-bit message blocks m_j and an $8n$-bit chaining value h_j as inputs, and produces an $8n$-bit output. The $8n$-bit hash value is computed using the compression function f:

$$\begin{aligned} h_0 &= IV \\ h_{j+1} &= f(h_j, m_j), \quad 0 \le j < t \\ hash &= h_t. \end{aligned}$$

Furthermore, NIST defines a general truncation procedure for SHA-256 and SHA-512, which includes SHA-256/224, SHA-512/224, SHA-512/256 and SHA-384.

The compression function f is computed through iterative updates to internal states. The number of steps, which is denoted by r, is 64 for SHA-256 and 80 for SHA-512, respectively. In the following, we provide a brief overview of the compression function f. It consists of two main parts: the message expansion and the state update transformation. A complete description of SHA-2 is given in [Nat12].

Message Expansion. The 512-bit message block for SHA-256 and the 1024-bit message block for SHA-512 are divided into 16 message words of sizes 32 bits and 64 bits, respectively,

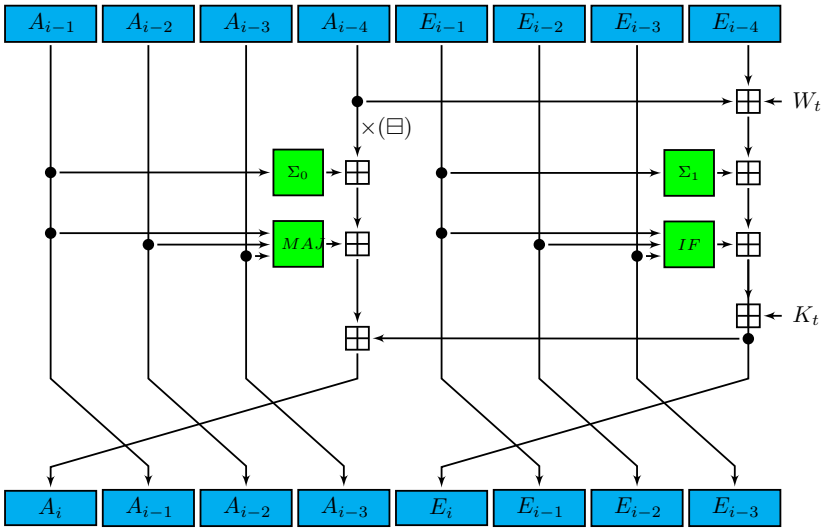

Figure 1: The state update transformation of SHA-2.

which are denoted by $(m_0, \ldots, m_{15})$. Then, the 16 message words are expanded to r expanded message words W_i, i.e., $W_0, W_1, \ldots, W_{r-1}$:

$$W_i = \begin{cases} m_i & 0 \le i \le 15, \\ \sigma_1(W_{i-2}) \boxplus W_{i-7} \boxplus \sigma_0(W_{i-15}) \boxplus W_{i-16} & 16 \le i \le r-1. \end{cases}$$

The functions $\sigma_0(x)$ and $\sigma_1(x)$ in SHA-256 are given by

$$\begin{aligned} \sigma_0(x) &= (x \ggg 7) \oplus (x \ggg 18) \oplus (x \gg 3), \\ \sigma_1(x) &= (x \ggg 17) \oplus (x \ggg 19) \oplus (x \gg 10). \end{aligned}$$

The functions $\sigma_0(x)$ and $\sigma_1(x)$ in SHA-512 are given by

$$\begin{aligned} \sigma_0(x) &= (x \ggg 1) \oplus (x \ggg 8) \oplus (x \gg 7), \\ \sigma_1(x) &= (x \ggg 19) \oplus (x \ggg 61) \oplus (x \gg 6). \end{aligned}$$

State update transformation. We utilize the alternate description for the state update of SHA-256 and SHA-512, as illustrated in Figure 1.

The state update transformation starts from an $8n$-bit chaining value

$$iv = (A_{-1}, \ldots, A_{-4}, E_{-1}, \ldots, E_{-4}),$$

and updates it by applying the step function r times. In each step $i \in \{0, \ldots, r-1\}$, one expanded message word W_i is used to compute the two internal state words E_i and A_i as follows, where K_i is a predefined constant and can be referred to [Nat12].

$$\begin{aligned} E_i &= A_{i-4} \boxplus E_{i-4} \boxplus \Sigma_1(E_{i-1}) \boxplus \mathrm{IF}(E_{i-1}, E_{i-2}, E_{i-3}) \boxplus K_i \boxplus W_i, \\ A_i &= E_i \boxminus A_{i-4} \boxplus \Sigma_0(A_{i-1}) \boxplus \mathrm{MAJ}(A_{i-1}, A_{i-2}, A_{i-3}). \end{aligned}$$

Both SHA-256 and SHA-512 utilize the same Boolean functions IF and MAJ, as defined below:

$$\begin{aligned} \mathrm{IF}(x, y, z) &= (x \wedge y) \oplus (x \wedge z) \oplus z, \\ \mathrm{MAJ}(x, y, z) &= (x \wedge y) \oplus (x \wedge z) \oplus (y \wedge z). \end{aligned}$$

However, the linear functions Σ_0 and Σ_1 are different for SHA-256 and SHA-512. For SHA-256, they are defined below:

$$\begin{aligned}\Sigma_0(x) &= (x \ggg 2) \oplus (x \ggg 13) \oplus (x \ggg 22),\\ \Sigma_1(x) &= (x \ggg 6) \oplus (x \ggg 11) \oplus (x \ggg 25).\end{aligned}$$

For SHA-512, they are defined below:

$$\begin{aligned}\Sigma_0(x) &= (x \ggg 28) \oplus (x \ggg 34) \oplus (x \ggg 39),\\ \Sigma_1(x) &= (x \ggg 14) \oplus (x \ggg 18) \oplus (x \ggg 41).\end{aligned}$$

After the last step of the state update transformation, the previous chaining value is added to the output of the state update. The result of this feed-forward sum is the chaining value h:

$$h = (A_{r-1} \boxplus A_{-1}, \ldots, A_{r-4} \boxplus A_{-4}, E_{r-1} \boxplus E_{-1}, \ldots, E_{r-4} \boxplus E_{-4}).$$

3 The First Collision Attack on 38-Step SHA-2

At CRYPTO 2026, after revisiting the constructions of local collisions in many collision attacks on SHA-2 in the literature, it was found that by injecting nonzero differences in

$$(W_{4+i}, \ldots, W_{8+i}, W_{12+i}, W_{13+i}, W_{20+i}, W_{22+i})_{0\le i\le 3},$$

it is possible to achieve collision attacks on $(i+35)$-step SHA-2 [LLWS26] under the memory-efficient collision attack framework developed at ASIACRYPT 2024 [LLW+24b]. However, although collision attacks on 35/36/37-step SHA-2 were all achieved [LLWS26], the authors could not find high-probability 38-step differential characteristics for either SHA-256 or SHA-512 to extend the attack to 38 steps.

As a natural extension of the work at CRYPTO 2026 [LLWS26], we address the issue to find high-probability 38-step differential characteristics that can lead to effective 38-step collision attacks. The main idea is to refine the search strategy without changing the strategy to construct local collisions, i.e., the nonzero message differences only exist in

$$(W_7, \ldots, W_{11}, W_{15}, W_{16}, W_{23}, W_{25})$$

for the 38-step differential characteristics. This new search procedure turns out to be very effective. As a result, we also apply the refined search strategy to finding improved 36/37-step differential characteristics to significantly improve the 36/37-step collision attacks.

3.1 Collision Attack on 38-Step SHA-2

We still use the open-source SAT/SMT-based tool [LLW24a] to search for SHA-2 differential characteristics. We refer the interested readers to [LLW24a] for more technical details of the SAT/SMT-based tool. The refined search procedure is stated below.

Step 1: **Minimize the Hamming weight of $(\nabla W_i)_{0\le i\le 37}$.** Specifically, find a solution of $(\nabla W_i)_{0\le i\le 37}$ with the minimal value of $\sum_{i=0}^{37} \mathbf{H}(\nabla W_i)$ where nonzero message differences only exist in $(W_7, \ldots, W_{11}, W_{15}, W_{16}, W_{23}, W_{25})$.

Step 2: **Restrict the Hamming weight of $(\nabla E_i)_{16\le i\le 25}$.** Specifically, with the fixed solution of $\sum_{i=0}^{37} \mathbf{H}(\nabla W_i)$, we aim to search for differential characteristic satisfying

$$\begin{aligned}\forall i \in [0,37] \text{ and } i \notin \{5,\ldots,14\} : \delta A_i &= 0,\\ \forall i \in [0,37] \text{ and } i \notin \{5,\ldots,19\} : \delta E_i &= 0,\end{aligned}$$

Our new insight for this step comes from the step update functions. Specifically, we can write the expressions of (E_{23}, E_{24}, E_{25}) and $(A_{19}, \ldots, A_{25})$ as follows, where state words colored red represent that their differences are nonzero.

$$\begin{aligned}
E_{25} &= A_{21} \boxplus E_{21} \boxplus \Sigma_1(E_{24}) \boxplus \mathrm{IF}(E_{24}, E_{23}, E_{22}) \boxplus K_{25} \boxplus W_{25},\\
E_{24} &= A_{20} \boxplus E_{20} \boxplus \Sigma_1(E_{23}) \boxplus \mathrm{IF}(E_{23}, E_{22}, E_{21}) \boxplus K_{24} \boxplus W_{24},\\
E_{23} &= A_{19} \boxplus E_{19} \boxplus \Sigma_1(E_{22}) \boxplus \mathrm{IF}(E_{22}, E_{21}, E_{20}) \boxplus K_{23} \boxplus W_{23},\\
A_{25} &= E_{25} \boxminus A_{21} \boxplus \Sigma_0(A_{24}) \boxplus \mathrm{MAJ}(A_{24}, A_{23}, A_{22}),\\
A_{24} &= E_{24} \boxminus A_{20} \boxplus \Sigma_0(A_{23}) \boxplus \mathrm{MAJ}(A_{23}, A_{22}, A_{21}),\\
A_{23} &= E_{23} \boxminus A_{19} \boxplus \Sigma_0(A_{22}) \boxplus \mathrm{MAJ}(A_{22}, A_{21}, A_{20}),\\
A_{22} &= E_{22} \boxminus A_{18} \boxplus \Sigma_0(A_{21}) \boxplus \mathrm{MAJ}(A_{21}, A_{20}, A_{19}),\\
A_{21} &= E_{21} \boxminus A_{17} \boxplus \Sigma_0(A_{20}) \boxplus \mathrm{MAJ}(A_{20}, A_{19}, A_{18}),\\
A_{20} &= E_{20} \boxminus A_{16} \boxplus \Sigma_0(A_{19}) \boxplus \mathrm{MAJ}(A_{19}, A_{18}, A_{17}),\\
A_{19} &= E_{19} \boxminus A_{15} \boxplus \Sigma_0(A_{18}) \boxplus \mathrm{MAJ}(A_{18}, A_{17}, A_{16}).
\end{aligned}$$

Note that the differences of (E_{23}, E_{24}, E_{25}) and $(A_{19}, \ldots, A_{25})$ are all required to be 0. Hence, δE_{21} mainly depends on δW_{25}, δE_{19} depends on δW_{23}, δA_{17} depends on δE_{21}, and δA_{15} depends on δE_{19}. Therefore, we set

$$\text{minimize } \mathcal{O}_1 = \sum_{i=16}^{19} \mathbf{H}(\nabla E_i)$$

as the objective function. Similar to Step 1, we only record the minimal value of $\mathcal{O}_1$ but will not fix the corresponding $(\nabla A_i, \nabla E_i)_{0 \le i \le 37}$.

Step 3: **Find the minimal Hamming weight of $(\nabla E_i)_{14 \le i \le 15}$.** Based on the solution of $\sum_{i=0}^{37} \mathbf{H}(\nabla W_i)$ obtained at Step 1, and the solution of $\mathcal{O}_1 = \sum_{i=16}^{19} \mathbf{H}(\nabla E_i)$ obtained at Step 2, we further search for a solution satisfying:

$$\begin{aligned}
\forall i \in [0, 37] \text{ and } i \notin \{5, \ldots, 14\} : \delta A_i &= 0,\\
\forall i \in [0, 37] \text{ and } i \notin \{5, \ldots, 19\} : \delta E_i &= 0,
\end{aligned}$$

where the objective function is set as

$$\text{minimize } \mathcal{O}_2 = \sum_{i=14}^{15} \mathbf{H}(\nabla E_i).$$

Here, we still only record $\mathcal{O}_2$, and will not store the corresponding solution of each ∇A_i and ∇E_i.

Step 4: **Find the minimal Hamming weight of $\mathbf{H}(\nabla A_i)_{5 \le i \le 25}$.** Similar to Step 3, we add one more constraint $\mathcal{O}_2 = \sum_{i=14}^{15} \mathbf{H}(\nabla E_i)$ to the model, and search for a required solution of $(\nabla A_i, \nabla E_i)_{0 \le i \le 37}$ where the objective function is set as

$$\text{minimize } \mathcal{O}_3 = \sum_{i=5}^{25} \mathbf{H}(\nabla A_i).$$

We are still only interested in the solution of $\mathcal{O}_3$ at this step.

Step 5: **Find the complete differential characteristic.** With the fixed solutions of $\sum_{i=0}^{37} \mathbf{H}(\nabla W_i)$, $\mathcal{O}_1$, $\mathcal{O}_2$ and $\mathcal{O}_3$, we search for a concrete required solution of

$$(\nabla A_i, \nabla E_i, \nabla W_i)_{0 \le i \le 37}$$

while still minimizing $\sum_{i=5}^{25} \mathbf{H}(\nabla E_i)$. If a solution is found, a candidate solution of the 38-step differential characteristic is found.

Search result. We successfully find a valid high-probability differential characteristic for 38-step SHA-256, as shown in Table 3. To demonstrate that the 38-step differential characteristic for SHA-256 is valid, we provide a SFS collision instance shown in Table 5.

Meet-in-the-middle technique to find collisions. In the 38-step differential characteristic shown in Table 3, there are no differences in the first 7 message words $(W_0, \ldots, W_6)$. Following the same idea as in [LLW24a, LLW+24b, LLW+25, LLWS26, MNS13], we convert the 38-step SFS collision attack into a collision attack using a two-block meet-in-the-middle method.

Different from the original memory-efficient collision attack framework proposed at ASIACRYPT 2024 [LLW+24b] where the attacker still needs to store some solutions of internal state words, an improved framework was proposed at CRYPTO 2026 [LLWS26] where almost no intermediate solutions are required to store. In other words, the meet-in-the-middle phase is completely on-the-fly in the improved collision attack framework [LLWS26]. However, we also want to mention that this new memory-efficient collision attack framework was only applied to the 35-step practical collision attacks in [LLWS26].

Based on the 38-step differential characteristic listed in Table 3, the detailed 38-step collision attack procedure under the improved framework [LLWS26] is stated below.

Step 1: Find a valid solution in the expanded message words $(W_i)_{8\le i\le 13}$, as well as the state words $(A_i)_{0\le i\le 13}$ and $(E_i)_{4\le i\le 13}$ by the step update function.

Step 2: Try an arbitrary M_0 and obtain the corresponding chaining input $CV_1 = (A_{-4}, \ldots, A_{-1}, E_{-4}, \ldots, E_{-1})$ and check the validity of A_{-1} on-the-fly. Specifically, based on the step update function, we can easily transform the validity check of A_{-1} into checking the conditions on W_7.

$$\begin{aligned} E_3 &= A_3 \boxplus A_{-1} \boxminus \Sigma_0(A_2) \boxminus \mathrm{MAJ}(A_2, A_1, A_0), \\ W_7 &= E_7 \boxminus A_3 \boxminus E_3 \boxminus \Sigma_1(E_6) \boxminus \mathrm{IF}(E_6, E_5, E_4) \boxminus K_7. \end{aligned}$$

When A_{-1} is valid, i.e., differential conditions on the corresponding W_7 hold, $(W_0, \ldots, W_6)$ can be computed as follows:

$$\begin{aligned} E_2 &= A_2 \boxplus A_{-2} \boxminus \Sigma_0(A_1) \boxminus \mathrm{MAJ}(A_1, A_0, A_{-1}), \\ W_6 &= E_6 \boxminus A_2 \boxminus E_2 \boxminus \Sigma_1(E_5) \boxminus \mathrm{IF}(E_5, E_4, E_3) \boxminus K_6, \end{aligned}$$

$$\begin{aligned} E_1 &= A_1 \boxplus A_{-3} \boxminus \Sigma_0(A_0) \boxminus \mathrm{MAJ}(A_0, A_{-1}, A_{-2}), \\ W_5 &= E_5 \boxminus A_1 \boxminus E_1 \boxminus \Sigma_1(E_4) \boxminus \mathrm{IF}(E_4, E_3, E_2) \boxminus K_5, \end{aligned}$$

$$\begin{aligned} E_0 &= A_0 \boxplus A_{-4} \boxminus \Sigma_0(A_{-1}) \boxminus \mathrm{MAJ}(A_{-1}, A_{-2}, A_{-3}), \\ W_4 &= E_4 \boxminus A_0 \boxminus E_0 \boxminus \Sigma_1(E_3) \boxminus \mathrm{IF}(E_3, E_2, E_1) \boxminus K_4, \end{aligned}$$

$$\begin{aligned} W_3 &= E_3 \boxminus A_{-1} \boxminus E_{-1} \boxminus \Sigma_1(E_2) \boxminus \mathrm{IF}(E_2, E_1, E_0) \boxminus K_3, \\ W_2 &= E_2 \boxminus A_{-2} \boxminus E_{-2} \boxminus \Sigma_1(E_1) \boxminus \mathrm{IF}(E_1, E_0, E_{-1}) \boxminus K_2, \\ W_1 &= E_1 \boxminus A_{-3} \boxminus E_{-3} \boxminus \Sigma_1(E_0) \boxminus \mathrm{IF}(E_0, E_{-1}, E_{-2}) \boxminus K_1, \\ W_0 &= E_0 \boxminus A_{-4} \boxminus E_{-4} \boxminus \Sigma_1(E_{-1}) \boxminus \mathrm{IF}(E_{-1}, E_{-2}, E_{-3}) \boxminus K_0. \end{aligned}$$

Step 3: Up to this step, $(W_i)_{0\le i\le 13}$ have been fixed. Use the degrees of freedom in $(W_i)_{14\le i\le 15}$ to fulfill the remaining uncontrolled conditions on $(A_i, E_i, W_i)_{i\ge 16}$. If it fails, go to Step 2.

Comparison to previous methods [LLW+24b, LLWS26]. From the shape of our 38-step differential characteristic, it can be observed that there are no conditions on (E_1, E_2, W_5, W_6). Hence, we only need to consider the probabilistic match in A_{-1} under the meet-in-the-middle frameworks [LLW+24b, LLWS26] to fulfill the differential conditions, which is equivalent to checking the conditions on W_7. The match in (A_{-2}, A_{-3}) holds with probability 1 by using all degrees of freedom in (W_5, W_6).

Complexity Evaluation. The time complexity of finding a valid solution in Step 1 using a SAT solver is approximately $2^{38.3}$, i.e., the time consumed is equal to about $2^{38.3}$ calls of the underlying compression function. In Step 2, there are 4 conditions on W_7. Hence, the probability of finding a valid M_0 is 2^{-4}. In Step 3, (W_{14}, W_{15}) has 2^2 possible values, and there are 102 conditions on $(A_i, E_i, W_i)_{16 \le i \le 37}$, as shown in Table 4. Therefore, $2^{102-2} = 2^{100}$ valid values of M_0 are required, and the corresponding time complexity of Step 2 is $2^{100} \cdot 2^4 = 2^{104}$. Hence, the total time complexity of the collision attack on 38-step SHA-256 is $2^{104} + 2^{102} \approx 2^{104.3}$.

3.2 Application to 38-Step SHA-512

SHA-256 and SHA-512 share a similar structure, with the primary difference being the size of the state words and the Boolean functions σ and Σ. We also applied the same procedure to search for a 38-step differential characteristic of SHA-512. The result is given in Table 6 and the corresponding SFS colliding message pair to verify its correctness is shown in Table 7. Applying the same meet-in-the-middle framework as in the above attack on 38-step SHA-256, we successfully achieve the first theoretical collision attack on 38-step SHA-512. The detailed evaluation of the time complexity of the collision attack on 38-step SHA-512 is given below.

Complexity Evaluation. The time complexity of finding a valid solution in Step 1 using a SAT solver is approximately $2^{38.9}$. In Step 2, there are 4 conditions on W_7. Hence, the probability of finding a valid M_0 is 2^{-4}. In Step 3, (W_{14}, W_{15}) has $2^{5.6}$ possible values, and there are 125 conditions on $(A_i, E_i, W_i)_{16 \le i \le 37}$, as shown in Table 8. Therefore, $2^{125-5.6} = 2^{119.4}$ valid values of M_0 are required, and the corresponding time complexity of Step 2 is $2^{119.4} \cdot 2^4 = 2^{123.4}$. Hence, the total time complexity of the collision attack on 38-step SHA-512 is $2^{123.4} + 2^{125} \approx 2^{125.4}$.

Remark 1. At the end of this section, we want to highlight the main contribution in the 38-step collision attacks compared to [LLW+24b, LLWS26]. In a word, a more refined search procedure is used in order to make the uncontrolled parts of the differential characteristic as sparse as possible under the memory-efficient collision attack framework. Specifically, instead of minimizing the Hamming weight of all $(\nabla A_i)_{0 \le i \le 37}$ at one step, we first minimize the Hamming weight of the signed differences of A_i in the uncontrolled part, and then further minimize the whole Hamming weight of $(\nabla A_i)_{0 \le i \le 37}$. Moreover, compared to previous memory-efficient collision attack frameworks [LLW+24b, LLWS26], since there are no conditions on $(E_1, E_2, E_3, W_5, W_6)$, we only need to consider the probabilistic match in A_{-1}, which is converted into checking conditions only on W_7. Hence, the match is completely on-the-fly and more memory-efficient, as long as the number of differential conditions on W_7 is small. Note that in the latest memory-efficient collision attack framework proposed for practical 35-step collision attacks [LLWS26], the state words (A_{-1}, A_{-2}, A_{-3}) are also matched completely on-the-fly, which is different from the previous one at ASIACRYPT 2024 [LLW+24b] where only (A_{-2}, A_{-3}) are matched on-the-fly. Therefore, the 38-step attack is a special case of the framework in [LLWS26] by exploiting the special shape of the 38-step differential characteristics. It is also interesting to find that as the number of attacked steps increases, the number of state words to be

probabilistically matched decreases. Indeed, this phenomenon can also be found in the subsequent improved 36/37-step collision attacks.

4 Generalizing the Search Procedure and Application to 36/37/39-Step SHA-2

The successful theoretical collision attacks on 38-step SHA-2 in Section 3 indicate that the new search strategy is significantly effective. Therefore, it is promising to improve the best 36/37-step collision attacks [LLWS26] by using the same search strategy to find improved differential characteristics. In this section, we describe the details of how to find improved 36/37/39-step differential characteristics of SHA-256 and SHA-512.

4.1 Improved 36/37-Step Differential Characteristics of SHA-2

As in [LLWS26], the nonzero message differences only exist in

$$(W_5, \ldots, W_9, W_{13}, W_{14}, W_{21}, W_{23})$$

for the 36-step differential characteristics. For the 37-step differential characteristics, the nonzero message differences only exist in

$$(W_6, \ldots, W_{10}, W_{14}, W_{15}, W_{22}, W_{24}).$$

In the following, we give a general description of the search procedure to find improved r-step differential characteristics of both SHA-256 and SHA-512 where $r \geq 36$.

Step 1: **Minimize the Hamming weight of $(\nabla W_i)_{0\leq i\leq r-1}$.** Specifically, find a solution of $(\nabla W_i)_{0\leq i\leq r-1}$ with the minimal value of $\sum_{i=0}^{r-1} \mathbf{H}(\nabla W_i)$ where nonzero message differences only exist in $(W_{r-31}, \ldots, W_{r-27}, W_{r-23}, W_{r-22}, W_{r-15}, W_{r-13})$.

Step 2: **Restrict the Hamming weight of $(\nabla E_i)_{16\leq i\leq r-13}$.** Specifically, with the fixed solution of $\sum_{i=0}^{r-1} \mathbf{H}(\nabla W_i)$, we aim to search for differential characteristic satisfying

$$\begin{aligned} \forall i \in [0, r-1] \text{ and } i \notin \{r-31, \ldots, r-22\} : \delta A_i &= 0, \\ \forall i \in [0, r-1] \text{ and } i \notin \{r-31, \ldots, r-17\} : \delta E_i &= 0, \end{aligned}$$

Our new insight for this step comes from the step update functions. Specifically, we can write the expressions of $(E_{r-13}, E_{r-14}, E_{r-15})$ and $(A_{r-19}, \ldots, A_{r-13})$ as follows, where state words colored red represent that their differences are nonzero.

$$\begin{aligned}
E_{r-13} &= A_{r-17} \boxplus E_{r-17} \boxplus \Sigma_1(E_{r-14}) \boxplus \mathrm{IF}(E_{r-14}, E_{r-15}, E_{r-16}) \boxplus K_{r-13} \boxplus W_{r-13}, \\
E_{r-14} &= A_{r-18} \boxplus E_{r-18} \boxplus \Sigma_1(E_{r-15}) \boxplus \mathrm{IF}(E_{r-15}, E_{r-16}, E_{r-17}) \boxplus K_{r-14} \boxplus W_{r-14}, \\
E_{r-15} &= A_{r-19} \boxplus E_{r-19} \boxplus \Sigma_1(E_{r-16}) \boxplus \mathrm{IF}(E_{r-16}, E_{r-17}, E_{r-18}) \boxplus K_{r-15} \boxplus W_{r-15}, \\
A_{r-13} &= E_{r-13} \boxminus A_{r-17} \boxplus \Sigma_0(A_{r-14}) \boxplus \mathrm{MAJ}(A_{r-14}, A_{r-15}, A_{r-16}), \\
A_{r-14} &= E_{r-14} \boxminus A_{r-18} \boxplus \Sigma_0(A_{r-15}) \boxplus \mathrm{MAJ}(A_{r-15}, A_{r-16}, A_{r-17}), \\
A_{r-15} &= E_{r-15} \boxminus A_{r-19} \boxplus \Sigma_0(A_{r-16}) \boxplus \mathrm{MAJ}(A_{r-16}, A_{r-17}, A_{r-18}), \\
A_{r-16} &= E_{r-16} \boxminus A_{r-20} \boxplus \Sigma_0(A_{r-17}) \boxplus \mathrm{MAJ}(A_{r-17}, A_{r-18}, A_{r-19}), \\
A_{r-17} &= E_{r-17} \boxminus A_{r-21} \boxplus \Sigma_0(A_{r-18}) \boxplus \mathrm{MAJ}(A_{r-18}, A_{r-19}, A_{r-20}), \\
A_{r-18} &= E_{r-18} \boxminus A_{r-22} \boxplus \Sigma_0(A_{r-19}) \boxplus \mathrm{MAJ}(A_{r-19}, A_{r-20}, A_{r-21}), \\
A_{r-19} &= E_{r-19} \boxminus A_{r-23} \boxplus \Sigma_0(A_{r-20}) \boxplus \mathrm{MAJ}(A_{r-20}, A_{r-21}, A_{r-22}).
\end{aligned}$$

Note that the differences of $(E_{r-13}, E_{r-14}, E_{r-15})$ and $(A_{r-19}, \ldots, A_{r-13})$ are all required to be 0. Hence, δE_{r-17} mainly depends on δW_{r-13}, δE_{r-19} depends on

δW_{r-15}, δA_{r-21} depends on δE_{r-17}, and δA_{r-23} depends on δE_{r-19}. Therefore, we set

$$\text{minimize } \mathcal{O}_1 = \sum_{i=16}^{r-19} \mathbf{H}(\nabla E_i)$$

as the objective function. Similar to Step 1, we only record the minimal value of $\mathcal{O}_1$ but will not fix the corresponding $(\nabla A_i, \nabla E_i)_{0 \le i \le r-1}$.

Step 3: **Find the minimal Hamming weight of $(\nabla E_i)_{14 \le i \le 15}$.** Based on the solution of $\sum_{i=0}^{r-1} \mathbf{H}(\nabla W_i)$ obtained at Step 1, and the solution of $\mathcal{O}_1 = \sum_{i=16}^{r-19} \mathbf{H}(\nabla E_i)$ obtained at Step 2, we further search for a solution satisfying:

$$\begin{aligned} \forall i \in [0, r-1] \text{ and } i \notin \{r-31, \ldots, r-22\} : \delta A_i &= 0, \\ \forall i \in [0, r-1] \text{ and } i \notin \{r-31, \ldots, r-17\} : \delta E_i &= 0, \end{aligned}$$

where the objective function is set as

$$\text{minimize } \mathcal{O}_2 = \sum_{i=14}^{15} \mathbf{H}(\nabla E_i).$$

Here, we still only record $\mathcal{O}_2$, and will not store the corresponding solution of each ∇A_i and ∇E_i.

Step 4: **Find the minimal Hamming weight of $\mathbf{H}(\nabla A_i)_{r-31 \le i \le r-11}$.** Similar to Step 3, we add one more constraint $\mathcal{O}_2 = \sum_{i=14}^{15} \mathbf{H}(\nabla E_i)$ to the model, and search for a required solution of $(\nabla A_i, \nabla E_i)_{0 \le i \le r-1}$ where the objective function is set as

$$\text{minimize } \mathcal{O}_3 = \sum_{i=r-31}^{r-11} \mathbf{H}(\nabla A_i).$$

We are still only interested in the solution of $\mathcal{O}_3$ at this step.

Step 5: **Find the complete differential characteristic.** With the fixed solutions of $\sum_{i=0}^{r-1} \mathbf{H}(\nabla W_i)$, $\mathcal{O}_1$, $\mathcal{O}_2$ and $\mathcal{O}_3$, we search for a concrete required solution of

$$(\nabla A_i, \nabla E_i, \nabla W_i)_{0 \le i \le r-1}$$

while still minimizing $\sum_{i=r-31}^{r-11} \mathbf{H}(\nabla E_i)$. If a solution is found, a candidate solution of the r-step differential characteristic is found.

Application to 36-step SHA-2. Based on the above steps, we found a new 36-step SHA-256 differential characteristic, as shown in Table 9. We also provide a message pair for a 36-step SHA-256 SFS collision, as shown in Table 11, to demonstrate the validity of the newly found differential characteristic. Similarly, the new 36-step SHA-512 differential characteristic and the corresponding SFS collision are given in Table 12 and Table 14, respectively.

Application to 37-step SHA-2. The new 37-step SHA-256 differential characteristic and the corresponding SFS collision are given in Table 15 and Table 17. The new 37-step SHA-512 differential characteristic and the corresponding SFS collision are given in Table 18 and Table 20, respectively.

4.2 Memory-Efficient Framework to Fulfill Differential Conditions

With improved differential characteristics at hand, the remaining work for finding collisions is relatively easy. Specifically, based on them, we can utilize the latest memory-efficient collision attack framework proposed at CRYPTO 2026 [LLWS26] to efficiently fulfill differential conditions, just as in the above 38-step attacks. Different from the 38-step collision attacks, the probabilistic meet-in-the-middle match is conducted in (A_{-1}, A_{-2}, A_{-3}) and (A_{-1}, A_{-2}) in the 36-step and 37-step collision attacks, respectively. The match is always completely on-the-fly and thus the 36/37-step collision attacks also consume much less memory compared to [LLWS26]. Note that in [LLWS26], such an improved memory-efficient framework was only applied to 35-step practical collision attacks.

We also give a general description of the memory-efficient collision attack framework [LLWS26] such that it can be applied to r steps of SHA-2 where $r > 35$.

Step 1: Find a valid solution in the expanded message words $(W_8, \ldots, W_{13})$, as well as the state words $(A_i)_{0\le i\le 13}$ and $(E_i)_{4\le i\le 13}$ by the step update function.

Step 2: Try an arbitrary M_0 and obtain the corresponding chaining input $CV_1 = (A_{-4}, \ldots, A_{-1}, E_{-4}, \ldots, E_{-1})$ and check the validity of (A_{-3}, A_{-2}, A_{-1}) on-the-fly. Specifically, based on the step update function, we can easily transform the validity check of (A_{-3}, A_{-2}, A_{-1}) into checking the conditions on (E_3, W_5, W_6, W_7).

$$\begin{aligned} E_3 &= A_3 \boxplus A_{-1} \boxminus \Sigma_0(A_2) \boxminus \mathrm{MAJ}(A_2, A_1, A_0), \\ W_7 &= E_7 \boxminus A_3 \boxminus E_3 \boxminus \Sigma_1(E_6) \boxminus \mathrm{IF}(E_6, E_5, E_4) \boxminus K_7. \end{aligned}$$

$$\begin{aligned} E_2 &= A_2 \boxplus A_{-2} \boxminus \Sigma_0(A_1) \boxminus \mathrm{MAJ}(A_1, A_0, A_{-1}), \\ W_6 &= E_6 \boxminus A_2 \boxminus E_2 \boxminus \Sigma_1(E_5) \boxminus \mathrm{IF}(E_5, E_4, E_3) \boxminus K_6. \end{aligned}$$

$$\begin{aligned} E_1 &= A_1 \boxplus A_{-3} \boxminus \Sigma_0(A_0) \boxminus \mathrm{MAJ}(A_0, A_{-1}, A_{-2}), \\ W_5 &= E_5 \boxminus A_1 \boxminus E_1 \boxminus \Sigma_1(E_4) \boxminus \mathrm{IF}(E_4, E_3, E_2) \boxminus K_5. \end{aligned}$$

When (A_{-1}, A_{-2}, A_{-3}) is valid, i.e., the corresponding (E_3, W_5, W_6, W_7) satisfy their differential conditions, based on the remaining state of the obtained CV_1 and the state of $(A_i)_{0\le i\le 13}$ and $(E_i)_{4\le i\le 13}$, the remaining message words $(W_0, \ldots, W_4)$ can be computed as follows.

$$\begin{aligned} E_0 &= A_0 \boxplus A_{-4} \boxminus \Sigma_0(A_{-1}) \boxminus \mathrm{MAJ}(A_{-1}, A_{-2}, A_{-3}), \\ W_4 &= E_4 \boxminus A_0 \boxminus E_0 \boxminus \Sigma_1(E_3) \boxminus \mathrm{IF}(E_3, E_2, E_1) \boxminus K_4, \end{aligned}$$

$$\begin{aligned} W_3 &= E_3 \boxminus A_{-1} \boxminus E_{-1} \boxminus \Sigma_1(E_2) \boxminus \mathrm{IF}(E_2, E_1, E_0) \boxminus K_3, \\ W_2 &= E_2 \boxminus A_{-2} \boxminus E_{-2} \boxminus \Sigma_1(E_1) \boxminus \mathrm{IF}(E_1, E_0, E_{-1}) \boxminus K_2, \\ W_1 &= E_1 \boxminus A_{-3} \boxminus E_{-3} \boxminus \Sigma_1(E_0) \boxminus \mathrm{IF}(E_0, E_{-1}, E_{-2}) \boxminus K_1, \\ W_0 &= E_0 \boxminus A_{-4} \boxminus E_{-4} \boxminus \Sigma_1(E_{-1}) \boxminus \mathrm{IF}(E_{-1}, E_{-2}, E_{-3}) \boxminus K_0. \end{aligned}$$

Step 3: At this step, $(W_i)_{0\le i\le 13}$ have been fixed. Use the degrees of freedom in $(W_i)_{14\le i\le 15}$ to fulfill the remaining uncontrolled conditions on $(A_i, E_i, W_i)_{i\ge 16}$. If it fails, go to Step 2.

Application to 36-step SHA-256. The time complexity of finding a valid solution in Step 1 using a SAT solver is approximately $2^{39.8}$, i.e., equal to $2^{39.8}$ calls of the SHA-256 compression function. In Step 2, there are 1, 4, 6 and 4 conditions on E_3, W_5, W_6 and W_7,

respectively. After testing 2^{40} random M_0, we found that the probability of finding a valid M_0 is approximately 2^{-15}. In Step 3, (W_{14}, W_{15}) has 2^{20} possible values, and there are 57 conditions on $(A_i, E_i, W_i)_{i\geq 16}$, as shown in Table 10. Therefore, $2^{57-20} = 2^{37}$ valid values of M_0 are required, and the corresponding time complexity is $2^{37} \cdot 2^{15} = 2^{52}$. Hence, the time complexity of the 36-step SHA-256 collision attack is $2^{52} + 2^{57} \approx 2^{57}$. We provide the first colliding message pair for 36-step SHA-256 in Table 2.

Table 2: The colliding message pair for 36 steps of SHA-256

M_0	d7c6ac05	3be3567b	33c02b26	3a6dfd82	ed46810e	8dacae35	3e004584	26d29992
	39cff659	0acc2223	8e77b3b0	cc7f4bba	9ff3abea	17606f86	ce693676	c3bc18aa
M_1	09abc425	0e8b8121	85808046	fadb1bca	394268e3	b9dfbd34	ae156845	74169a81
	1ea03337	a3210f16	79b82017	91059d10	97294bab	65ceec9c	89c67ae2	ac5eb7f9
M_1'	09abc425	0e8b8121	85808046	fadb1bca	394268e3	99dfbd34	aa556045	54169a81
	1fa123bd	a3290316	79b82017	91059d10	97294bab	618ee49c	a9c67ae2	ac5eb7f9
hash	79df3078	14339285	d39ae368	142866bc	ba063acf	fa4431c6	a5d74825	c4be00db

Application to 36-step SHA-512. The time complexity of finding a valid solution in Step 1 using a SAT solver is approximately $2^{39.4}$. In Step 2, there are 4, 4, 6 and 4 conditions on E_3, W_5, W_6 and W_7, respectively. After testing 2^{40} random M_0, we found that the probability of finding a valid M_0 is approximately 2^{-18}. In Step 3, (W_{14}, W_{15}) has 2^{21} possible values, and there are 66 conditions on $(A_i, E_i, W_i)_{i\geq 16}$, as shown in Table 13. Therefore, $2^{66-21} = 2^{45}$ valid values of M_0 are required, and the corresponding time complexity is $2^{45} \cdot 2^{18} = 2^{63}$. Hence, the time complexity of the 36-step SHA-512 collision attack is $2^{66} + 2^{63} \approx 2^{66.2}$.

Application to 37-step SHA-2. We note that in the 37-step differential characteristics of SHA-256 and SHA-512 displayed in Table 15 and Table 18, respectively, there are no conditions on (E_3, W_5). As a result, in Step 2, we only need to check the conditions on (W_6, W_7) in order to match (A_{-1}, A_{-2}).

For SHA-256, the time complexity of finding a valid solution in Step 1 using a SAT solver is approximately $2^{41.3}$. In Step 2, W_6 and W_7 have 3 and 6 conditions, respectively. After testing 2^{30} random M_0, we found that the probability of finding a valid M_0 is approximately 2^{-9}. In Step 3, (W_{14}, W_{15}) has 2^5 possible values, and there are 75 conditions on $(A_i, E_i, W_i)_{i\geq 16}$, as shown in Table 16. Therefore, $2^{75-5} = 2^{70}$ valid values of M_0 are required, and the corresponding time complexity is $2^{70} \cdot 2^9 = 2^{79}$. Hence, the time complexity of the 37-step SHA-256 collision attack is $2^{79} + 2^{75} \approx 2^{79.1}$.

For SHA-512, the time complexity of finding a valid solution in Step 1 using a SAT solver is approximately $2^{37.9}$. In Step 2, W_6 and W_7 have 3 and 6 conditions, respectively. After testing 2^{30} random M_0, we found that the probability of finding a valid M_0 is approximately 2^{-10}. In Step 3, (W_{14}, W_{15}) has $2^{14.6}$ possible values, and there are 87 conditions on $(A_i, E_i, W_i)_{i\geq 16}$, as shown in Table 19. Therefore, $2^{87-14.6} = 2^{62.4}$ valid values of M_0 are required, and the corresponding time complexity is $2^{62.4} \cdot 2^{10} = 2^{72.4}$. Hence, the time complexity of the 37-step SHA-512 collision attack is $2^{72.4} + 2^{87} \approx 2^{87}$.

4.3 Extended to 39 Steps

Indeed, the above 36/37/38-step collision attacks are all based on a similar idea to construct local collisions as in the SFS collision attack on 39-step SHA-512 [DEM15]. Based on our new findings for the corresponding differential characteristics and the meet-in-the-middle technique to fulfill the corresponding differential conditions, it seems that a 39-step collision

attack is also promising. Specifically, as in [DEM15], we can inject message differences in

$$(W_8, \ldots, W_{12}, W_{16}, W_{17}, W_{24}, W_{26}),$$

which is shifted for 1 step compared to the 38-step local collision. Note that in the 38-step collision attack, the message differences are injected in

$$(W_7, \ldots, W_{11}, W_{15}, W_{16}, W_{23}, W_{25}).$$

The shape of the differential characteristics are basically the same as in the 38-step ones. Moreover, there are not any differential conditions on

$$(W_5, W_6, W_7, E_1, E_2, E_3)$$

for such a local collision construction. This indicates that Step 2 of the memory-efficient framework to fulfill differential conditions always succeeds with probability 1. Hence, following the meet-in-the-middle memory-efficient framework in Section 4.2, as long as we can find a valid differential characteristic and the conditions after (A_{16}, E_{16}, W_{16}) hold with probability larger than $2^{-n/2}$ where $n = 512$ for SHA-512 and $n = 256$ for SHA-256, the collision attack can reach 39 steps.

Finding 39-step differential characteristics of SHA-2. We apply the general procedure described at Section 4.1 to search for 39-step SHA-2 differential characteristics. The results for SHA-256 and SHA-512 are shown in Table 21 and Table 24, respectively. To show their validity, we also provide the corresponding SFS collisions in Table 23 and Table 26, respectively.

Complexity evaluation for collision attack on 39-step SHA-256. In the memory-efficient framework stated in Section 4.2, Step 2 holds with probability 1 since there are no conditions on $(W_5, W_6, W_7, E_1, E_2, E_3)$. The time complexity of finding a valid solution in Step 1 using a SAT solver is approximately $2^{34.2}$. In Step 3, there are 168 conditions on $(A_i, E_i, W_i)_{i\geq 16}$, as shown in Table 22. Hence, the time complexity of the 39-step SHA-256 collision attack is 2^{168}, which is worse than the generic birthday attack. Therefore, this is an ineffective collision attack on 39-step SHA-256.

Complexity evaluation for 39-step SHA-512. We still use the same memory-efficient framework in Section 4.2 where Step 2 now holds with probability 1. For SHA-512, the time complexity of finding a valid solution in Step 1 using a SAT solver is approximately $2^{34.1}$. In Step 3, there are 178 conditions on $(A_i, E_i, W_i)_{i\geq 16}$, as shown in Table 24 and Table 25. Hence, the time complexity of the 39-step SHA-512 collision attack is 2^{178}. Therefore, we achieve the first collision attack on 39-step SHA-512.

5 Conclusion

In a latest research on collision attacks on round-reduced SHA-2 published at CRYPTO 2026, although 35/36/37-step collision attacks were all achieved, the 38-step collision attack is still out of reach. The bottleneck of the 38-step attack is the low probability of the uncontrolled part (i.e., after the 16-th step) of the 38-step differential characteristics. Hence, finding high-probility 38-step differential characteristics becomes the key to achieve the 38-step collision attack. We have successfully addressed this issue by proposing an improved search strategy to find SHA-2 differential characteristics based on the open-source SAT-based automatic tool for SHA-2. As a result, we achieve the first theoretical collision attacks on 38-step SHA-256 and SHA-512. Moreover, by searching for new 36/37-step

differential characteristics based on the same search strategy, the collision attacks on 36/37-step SHA-256 and SHA-512 are also significantly improved. In particular, based on the new 36-step differential characteristic, we found the first practical collision for 36-step SHA-256. Finally, we also apply our new search method to 39-step SHA-2. While the 39-step collision attack remains ineffective for SHA-256, the first effective collision attack on 39-step SHA-512 is achieved, marking a new milestone in collision attacks on the SHA-2 family.

References

[Dam89] Ivan Damgård. A design principle for hash functions. In Gilles Brassard, editor, *Advances in Cryptology - CRYPTO '89, 9th Annual International Cryptology Conference, Santa Barbara, California, USA, August 20-24, 1989, Proceedings*, Lecture Notes in Computer Science, pages 416–427. Springer, 1989.

[DEM15] Christoph Dobraunig, Maria Eichlseder, and Florian Mendel. Analysis of SHA-512/224 and SHA-512/256. In Tetsu Iwata and Jung Hee Cheon, editors, *Advances in Cryptology - ASIACRYPT 2015 - 21st International Conference on the Theory and Application of Cryptology and Information Security, Auckland, New Zealand, November 29 - December 3, 2015, Proceedings, Part II*, Lecture Notes in Computer Science, pages 612–630. Springer, 2015.

[EMS14] Maria Eichlseder, Florian Mendel, and Martin Schläffer. Branching heuristics in differential collision search with applications to SHA-512. In Carlos Cid and Christian Rechberger, editors, *Fast Software Encryption - 21st International Workshop, FSE 2014, London, UK, March 3-5, 2014. Revised Selected Papers*, Lecture Notes in Computer Science, pages 473–488. Springer, 2014.

[IMPR08] Sebastiaan Indesteege, Florian Mendel, Bart Preneel, and Christian Rechberger. Collisions and other non-random properties for step-reduced SHA-256. In Roberto Maria Avanzi, Liam Keliher, and Francesco Sica, editors, *Selected Areas in Cryptography, 15th International Workshop, SAC 2008, Sackville, New Brunswick, Canada, August 14-15, Revised Selected Papers*, volume 5381 of *Lecture Notes in Computer Science*, pages 276–293. Springer, 2008.

[LLW23] Yingxin Li, Fukang Liu, and Gaoli Wang. Automating collision attacks on RIPEMD-160. *IACR Trans. Symmetric Cryptol.*, 2023(4):112–142, 2023.

[LLW24a] Yingxin Li, Fukang Liu, and Gaoli Wang. New records in collision attacks on SHA-2. In Marc Joye and Gregor Leander, editors, *Advances in Cryptology - EUROCRYPT 2024 - 43rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zurich, Switzerland, May 26-30, 2024, Proceedings, Part I*, Lecture Notes in Computer Science, pages 158–186. Springer, 2024.

[LLW+24b] Yingxin Li, Fukang Liu, Gaoli Wang, Xiaoyang Dong, and Siwei Sun. The first practical collision for 31-step SHA-256. In Kai-Min Chung and Yu Sasaki, editors, *Advances in Cryptology - ASIACRYPT 2024 - 30th International Conference on the Theory and Application of Cryptology and Information Security, Kolkata, India, December 9-13, 2024, Proceedings, Part VII*, Lecture Notes in Computer Science, pages 237–266. Springer, 2024.

[LLW+25] Yingxin Li, Fukang Liu, Gaoli Wang, Haifeng Qian, Keting Jia, and Xiangyu Kong. New collision attacks on round-reduced SHA-512. In Yael Tauman

Kalai and Seny F. Kamara, editors, *Advances in Cryptology - CRYPTO 2025 - 45th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2025, Proceedings, Part V*, Lecture Notes in Computer Science, pages 200–229. Springer, 2025.

[LLWS26] Yingxin Li, Fukang Liu, Gaoli Wang, and Jiali Shi. Pushing the limit of memory-efficient collision attack framework for SHA-2. Cryptology ePrint Archive, Paper 2026/1080, 2026.

[LMS+22] Fukang Liu, Willi Meier, Santanu Sarkar, Gaoli Wang, Ryoma Ito, and Takanori Isobe. New cryptanalysis of ZUC-256 initialization using modular differences. *IACR Trans. Symmetric Cryptol.*, 2022(3):152–190, 2022.

[LWS+23] Fukang Liu, Gaoli Wang, Santanu Sarkar, Ravi Anand, Willi Meier, Yingxin Li, and Takanori Isobe. Analysis of RIPEMD-160: new collision attacks and finding characteristics with MILP. In Carmit Hazay and Martijn Stam, editors, *Advances in Cryptology - EUROCRYPT 2023 - 42nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Lyon, France, April 23-27, 2023, Proceedings, Part IV*, Lecture Notes in Computer Science, pages 189–219. Springer, 2023.

[Mer89] Ralph C. Merkle. One way hash functions and DES. In Gilles Brassard, editor, *Advances in Cryptology - CRYPTO '89, 9th Annual International Cryptology Conference, Santa Barbara, California, USA, August 20-24, 1989, Proceedings*, Lecture Notes in Computer Science, pages 428–446. Springer, 1989.

[MNS11] Florian Mendel, Tomislav Nad, and Martin Schläffer. Finding SHA-2 characteristics: Searching through a minefield of contradictions. In Dong Hoon Lee and Xiaoyun Wang, editors, *Advances in Cryptology - ASIACRYPT 2011 - 17th International Conference on the Theory and Application of Cryptology and Information Security, Seoul, South Korea, December 4-8, 2011. Proceedings*, Lecture Notes in Computer Science, pages 288–307. Springer, 2011.

[MNS12] Florian Mendel, Tomislav Nad, and Martin Schläffer. Collision attacks on the reduced dual-stream hash function RIPEMD-128. In Anne Canteaut, editor, *Fast Software Encryption - 19th International Workshop, FSE 2012, Washington, DC, USA, March 19-21, 2012. Revised Selected Papers*, volume 7549 of *Lecture Notes in Computer Science*, pages 226–243. Springer, 2012.

[MNS13] Florian Mendel, Tomislav Nad, and Martin Schläffer. Improving local collisions: New attacks on reduced SHA-256. In Thomas Johansson and Phong Q. Nguyen, editors, *Advances in Cryptology - EUROCRYPT 2013, 32nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Athens, Greece, May 26-30, 2013. Proceedings*, Lecture Notes in Computer Science, pages 262–278. Springer, 2013.

[Nat12] National Institute of Standards and Technology. Fips 180-4: Secure hash standard (shs). *Federal Information Processing Standards Publication 180-4, U.S. Department of Commerce*, 2012.

[SBK+17] Marc Stevens, Elie Bursztein, Pierre Karpman, Ange Albertini, and Yarik Markov. The first collision for full SHA-1. In Jonathan Katz and Hovav Shacham, editors, *Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part I*, volume 10401 of *Lecture Notes in Computer Science*, pages 570–596. Springer, 2017.

[SS08] Somitra Kumar Sanadhya and Palash Sarkar. New collision attacks against up to 24-step SHA-2. In Dipanwita Roy Chowdhury, Vincent Rijmen, and Abhijit Das, editors, *Progress in Cryptology - INDOCRYPT 2008, 9th International Conference on Cryptology in India, Kharagpur, India, December 14-17, 2008. Proceedings*, volume 5365 of *Lecture Notes in Computer Science*, pages 91–103. Springer, 2008.

[WLF+05] Xiaoyun Wang, Xuejia Lai, Dengguo Feng, Hui Chen, and Xiuyuan Yu. Cryptanalysis of the hash functions MD4 and RIPEMD. In Ronald Cramer, editor, *Advances in Cryptology - EUROCRYPT 2005, 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, May 22-26, 2005, Proceedings*, volume 3494 of *Lecture Notes in Computer Science*, pages 1–18. Springer, 2005.

[WY05] Xiaoyun Wang and Hongbo Yu. How to break MD5 and other hash functions. In Ronald Cramer, editor, *Advances in Cryptology - EUROCRYPT 2005, 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, May 22-26, 2005, Proceedings*, volume 3494 of *Lecture Notes in Computer Science*, pages 19–35. Springer, 2005.

[WYY05a] Xiaoyun Wang, Yiqun Lisa Yin, and Hongbo Yu. Finding collisions in the full SHA-1. In Victor Shoup, editor, *Advances in Cryptology - CRYPTO 2005: 25th Annual International Cryptology Conference, Santa Barbara, California, USA, August 14-18, 2005, Proceedings*, volume 3621 of *Lecture Notes in Computer Science*, pages 17–36. Springer, 2005.

[WYY05b] Xiaoyun Wang, Hongbo Yu, and Yiqun Lisa Yin. Efficient collision search attacks on SHA-0. In Victor Shoup, editor, *Advances in Cryptology - CRYPTO 2005: 25th Annual International Cryptology Conference, Santa Barbara, California, USA, August 14-18, 2005, Proceedings*, volume 3621 of *Lecture Notes in Computer Science*, pages 1–16. Springer, 2005.

[ZMLM26] Zhang Zhuolong, Li Muzhou, Gao Lei, and Wang Meiqin. Collision attacks on SHA-256 up to 37 steps with improved trail search. In Daemen Joan and Thomé Emmanuel, editors, *Advances in Cryptology - EUROCRYPT 2026 - 45th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Rome, Italy, May 10–14, 2026. Proceedings*, Lecture Notes in Computer Science, pages 91–120. Springer, 2026.

A Tables

Table 3: The differential characteristic for 38-step SHA-256

i	∇A_i	∇E_i	∇W_i
-4	================================	================================	
-3	================================	================================	
-2	================================	================================	
-1	================================	================================	
0	================================	================================	================================
1	================================	================================	================================
2	================================	================================	================================
3	================================	================================	================================
4	================================	================================	================================
5	================================	+++==============================	================================
6	================================	+++=1====0==+1=====0+===1==1====	================================
7	=nu=============================	uuu=0+1=11=0+00====1+===0==00=01	==n=============================
8	=========n=====n====n======u====	100=u+010n=1nu01=01nu=00n11u1=01	=====u===u==========n==========
9	================================	11000u1=n11u0000101101110=01u0uu	==u=============================
10	================================	==1010=01u00001u=010u=101==10111	=====n=u=======n===n==n=n=n=u=u=
11	==================u=======un==	1=n000=0111100101unn00011+111u10	===========u======u=u==========
12	=u===n====n======u===nu=======n=	01nuuu=110n111nu000100100+010u1n	================================
13	==n===========n================	00110101110=000u00111nuu+nnnu1n1	================================
14	====u=========nn========u==u====	=010n0===00===1n=11+0100+1110001	================================
15	======n=u=======n===============	=1==1=1==0=1==11===+u011n1011=1=	=====u===n==========n==========
16	================================	=u==10===n===+u1===n0=u=1==+====	==u=============================
17	==u=============================	=0=====+00===+0=+==01=0=0==+====	================================
18	================================	=0=====+01===n1=+==1==1===0u====	================================
19	================================	==11===nn====1==n===010===11==0=	================================
20	================================	==1====00====1==0==========1====	================================
21	================================	==u====10=======1===10==========	================================
22	================================	==0=============================	================================
23	================================	==1=============================	=====1=uu=====1=u=1=============
24	================================	================================	================================
25	================================	================================	==n=============================
26	================================	================================	================================
27	================================	================================	================================
28	================================	================================	================================
29	================================	================================	================================
30	================================	================================	================================
31	================================	================================	================================
32	================================	================================	================================
33	================================	================================	================================
34	================================	================================	================================
35	================================	================================	================================
36	================================	================================	================================
37	================================	================================	================================

Table 4: Extra conditions influencing the attack for the 38-step SHA-256 differential characteristic

	Conditions	Pro.
A_{16}	$A_{14}[15] = A_{16}[15], A_{14}[23] = A_{16}[23], A_{14}[25] = A_{16}[25], A_{15}[\ 4] = A_{16}[\ 4]$ $A_{15}[\ 7] = A_{16}[\ 7], A_{15}[16] \neq A_{16}[16], A_{15}[17] = A_{16}[17], A_{15}[27] = A_{16}[27]$ $A_{15}[29] = A_{16}[29]$	2^{-9}
A_{17}	$A_{16}[15] = A_{17}[15], A_{16}[23] = A_{17}[23], A_{16}[25] = A_{17}[25], A_{17}[\ 9] = A_{17}[20]$ $A_{17}[\ 6] = A_{17}[18], A_{17}[\ 8] = A_{17}[17]$	2^{-6}
A_{18}	$A_{16}[29] = A_{18}[29]$	2^{-1}
A_{19}	$A_{18}[29] = A_{19}[29]$	2^{-1}
E_{16}	$E_{16}[\ 4] \neq E_{16}[23], E_{16}[\ 3] \neq E_{16}[\ 8], E_{16}[14] = E_{16}[28]$	2^{-3}
E_{17}	$E_{16}[\ 4] = E_{17}[\ 4], E_{16}[18] = E_{17}[18]$	2^{-2}
E_{18}	$E_{18}[\ 0] \neq E_{18}[13], E_{17}[15] = E_{18}[15], E_{17}[24] = E_{18}[24]$	2^{-3}
E_{19}	$E_{19}[\ 6] \neq E_{19}[19], E_{19}[20] = E_{19}[\ 2]$	2^{-2}
E_{21}	$E_{21}[\ 2] = E_{21}[16]$	2^{-1}
W_7	$W_7[\ 8] \neq W_7[25], W_7[14] \neq W_7[18], W_7[\ 1] = W_7[12]$	2^{-3}
W_8	$W_8[\ 0] \neq W_8[28], W_8[30] \neq W_8[\ 9], W_8[\ 1] = W_8[18]$	2^{-3}
W_{16}	$W_{16}[\ 1] \neq W_{16}[12], W_{16}[20] \neq W_{16}[27], W_{16}[\ 8] = W_{16}[25], W_{16}[14] = W_{16}[18]$ $W_{16}[\ 4] \neq W_{16}[\ 6], W_{16}[22] \neq W_{16}[31]$	2^{-6}
W_{23}	$W_{23}[\ 0] \neq W_{23}[30], W_{23}[\ 1] \neq W_{23}[31], W_{23}[14] = W_{23}[21], W_{23}[16] = W_{23}[25]$	2^{-4}
W_{25}	$W_{25}[\ 4] = W_{25}[\ 9], W_{25}[22] = W_{25}[31], W_{25}[20] = W_{25}[27]$	2^{-3}

Table 5: The SFS colliding message pair for 38 steps of SHA-256

CV	cd278980	1b12a052	b87cc8a6	a9e059c5	c9c3db85	6ca4b5b5	63d13ac1	c0329f1e
M	48fc271b	9fca20cd	cc89f96f	fc40396f	8b328cb4	6b91ef78	97f9b767	e2450045
	3b016f58	de9804cb	66a99ea5	0ce30b8d	a28cd15a	77a1e994	d28e48a0	9b5f6dbb
M'	48fc271b	9fca20cd	cc89f96f	fc40396f	8b328cb4	6b91ef78	97f9b767	c2450045
	3f416758	fe9804cb	63a88c0f	0ceb1f8d	a28cd15a	77a1e994	d28e48a0	9f1f65bb
hash	5d9ca5f4	59ace3a3	26c9c26c	4252c585	4c0803b7	1b4d5ccd	25c3ccc0	90645c4d

Table 6: The differential characteristic for 38-step SHA-512

i	∇A_i	∇E_i	∇W_i
-4	==========	==========	
-3	==========	==========	
-2	==========	==========	
-1	==========	==========	
0	==========	==========	==========
1	==========	==========	==========
2	==========	==========	==========
3	==========	==========	==========
4	==========	==========	==========
5	==========	+==========	==========
6	==========	+0+000001+10++=+0++==========00==========	==========
7	u==========	n1+011010+01++=+0++====0==001====1==0===11===1=1+====1==11======	n==========0======
8	=n======u=====n===n=========u=====u====u=n==========	0nunnnnnnnuuuu1uuuu01=100010111010000000nu01=0=0+==1=0=001+01111	=n======u==========0=====
9	==========u==========	0u0n11uuu11n0110u0001=0000unn1001n00u0101n10=u=nu0=01n=1nn+10100	u==========0======
10	==========	0n11u01100101101011=00nuuuuuun0=001111=1000=01=010=001=10nunnnnn	=====n=n======n=====u======n==========nu=
11	====nn=====n======u==========n=====n=====n====n==========	u11n1u1100nu10u0un1100001011n10100nnu00011u00100n1101n1000010110	======u==========n==========u==
12	==========u======uu===u=======n===un====u==u==u=====uu======	0u001101u011101=10n10uu001uu10111u01u001101=u0000n0u0n1u00101101	==========
13	n==========u==========	0uuuuuuuuuuuuuu=00unn00uuu1n0u0uuu1nn10n00110unnn1nnn101=n00=0n0	==========
14	==========u===u=========n=====u=====uu==========	==011000=001111===100110000110100=111==n===+1111010000=1=1====0=	==========
15	==u=====u=======u==========	=10001001000010===0101000011u0000001n===0==+=00100101==0=1====1=	=u======n==========
16	==========	==========+=0=0=1uu===u=1=n===n=0==1=+=n===0=u=======n=0====	n==========0======
17	n==========	==+=====+=====+=1=1==00===0=0=0===1=1====+=0=====0=======0======	==========
18	==========	==+=====+=====u=0=u==11===1===1===1======n=1=====1=======1=1====	==========
19	==========	+=u=====u=====0=u=0==========0==========	==========
20	==========	+=0=====0=====1=0=1==========1==========	==========
21	==========	u=1=====1=======1==========	==========
22	==========	0==========	==========
23	==========	1==========	==n=====n=======n==========0==========
24	==========	==========	==========
25	==========	==========	n==========
26	==========	==========	==========
27	==========	==========	==========
28	==========	==========	==========
29	==========	==========	==========
30	==========	==========	==========
31	==========	==========	==========
32	==========	==========	==========
33	==========	==========	==========
34	==========	==========	==========
35	==========	==========	==========
36	==========	==========	==========
37	==========	==========	==========

Table 7: The SFS colliding message pair for 38 steps of SHA-512

CV	d0f04265cf36aac6	c3ec462db441213c	a86a0e0ba9be1e5e	f9710fe327a460ca
	cdf80e2c774096d6	06c0e2ae9704dcab	4608d059f5f09ce4	e28b4eb1b6492f16
M	5f222feae0b25ff0	bae990e0d75548f9	1389108d6998635e	d9fee20b874d2084
	66096b3fcacd185e	9e2076078ac4c635	f8fed73f88217956	ff5c0fa6ebdb93b8
	466f164e88de520a	5d9f7d293ee02381	9fee10b00626ebfd	6101b49e08d7b0e8
	7add8bbf53af82c9	0aa03bb55b9f3924	ef4ef863295d37ed	b9f1eb72db5befcd
M'	5f222feae0b25ff0	bae990e0d75548f9	1389108d6998635e	d9fee20b874d2084
	66096b3fcacd185e	9e2076078ac4c635	f8fed73f88217956	7f5c0fa6ebdb93b8
	06ef164e88de520a	dd9f7d293ee02381	9aec18a00626ebfb	6301a49e08d7b0ec
	7add8bbf53af82c9	0aa03bb55b9f3924	ef4ef863295d37ed	f971eb72db5befcd
hash	d4c9f5fd99a8f88a	091e2c6845e74e85	b605573df4e49e90	bb2d74b444eaaf10
	1b052eb73ee25b7b	b00bf176413c3e7e	feb4365298e0f18c	d2545d6c57e6c8d7

Table 8: Extra conditions influencing the attack for the 38-step SHA-512 differential characteristic

	Conditions	Pro.
A_{16}	$A_{14}[47] = A_{16}[47], A_{14}[55] = A_{16}[55], A_{14}[61] = A_{16}[61], A_{15}[22] = A_{16}[22]$ $A_{15}[23] = A_{16}[23], A_{15}[29] = A_{16}[29], A_{15}[35] = A_{16}[35], A_{15}[45] = A_{16}[45]$ $A_{15}[49] \neq A_{16}[49], A_{15}[63] \neq A_{16}[63]$	2^{-10}
A_{17}	$A_{16}[47] \neq A_{17}[47], A_{16}[55] \neq A_{17}[55], A_{16}[61] \neq A_{17}[61], A_{17}[52] \neq A_{17}[58]$ $A_{17}[57] \neq A_{17}[\ 4], A_{17}[\ 5] = A_{17}[10]$	2^{-6}
A_{18}	$A_{16}[63] = A_{18}[63]$	2^{-1}
A_{19}	$A_{18}[63] = A_{19}[63]$	2^{-1}
E_{16}	$E_{16}[57] = E_{16}[61], E_{16}[14] = E_{16}[51], E_{16}[\ 5] = E_{16}[46], E_{16}[\ 1] = E_{16}[38]$ $E_{16}[25] = E_{16}[52], E_{16}[15] \neq E_{16}[19]$	2^{-6}
E_{17}	$E_{16}[49] = E_{17}[49], E_{16}[22] = E_{17}[22]$	2^{-2}
E_{18}	$E_{18}[59] = E_{18}[63], E_{18}[12] \neq E_{18}[53], E_{17}[61] = E_{18}[61], E_{17}[55] = E_{18}[55]$	2^{-4}
E_{19}	$E_{19}[\ 1] = E_{19}[24], E_{19}[20] \neq E_{19}[57], E_{19}[18] = E_{19}[59], E_{19}[14] \neq E_{19}[51]$ $E_{19}[10] = E_{19}[51], E_{19}[\ 6] = E_{19}[43], E_{19}[34] = E_{19}[38], E_{19}[28] \neq E_{19}[32]$ $E_{19}[20] = E_{19}[24]$	2^{-9}
E_{21}	$E_{21}[\ 3] \neq E_{21}[26], E_{21}[22] \neq E_{21}[59], E_{21}[36] = E_{21}[40]$	2^{-3}
W_7	$W_7[\ 0] \neq W_7[57], W_7[56] = W_7[62]$	2^{-2}
W_8	$W_8[56] = W_8[63], W_8[49] \neq W_8[56], W_8[48] = W_8[54]$	2^{-3}
W_{16}	$W_{16}[12] = W_{16}[54], W_{16}[41] \neq W_{16}[50], W_{16}[8] = W_{16}[21], W_{16}[0] \neq W_{16}[57]$ $W_{16}[56] = W_{16}[62]$	2^{-5}
W_{23}	$W_{23}[10] = W_{23}[52], W_{23}[\ 5] = W_{23}[56], W_{23}[\ 4] \neq W_{23}[46], W_{23}[39] \neq W_{23}[48]$ $W_{23}[38] = W_{23}[60], W_{23}[33] = W_{23}[42], W_{23}[25] = W_{23}[34], W_{23}[\ 6] \neq W_{23}[19]$	2^{-8}
W_{25}	$W_{25}[12] = W_{25}[54], W_{25}[41] \neq W_{25}[50], W_{25}[8] = W_{25}[21]$	2^{-3}

Table 9: The differential characteristic for 36-step SHA-256

i	∇A_i	∇E_i	∇W_i
-4	================================	================================	
-3	================================	================================	
-2	================================	================================	
-1	================================	================================	
0	================================	================================	================================
1	================================	================================	================================
2	================================	================================	================================
3	================================	==1=============================	================================
4	================================	==0==1==0=00++======+======1====	================================
5	nuu=============================	==n==0==0011++1===0=+====1=0==11	==n=============================
6	=====u===n=====u=====n==n==n====	11u0=n11n1uuuu101000n1==100n0100	=====n===u==========n==========
7	================================	10110111nu0100u=11u10101=n001=uu	==n=============================
8	================================	=10==11=01000101==0001==10=1==00	=======u=======u===n====u=1=u=n=
9	================n=u==========	1nu110001001==00==0n0n==01110110	==========u=======nn==========
10	====n=u===n==========u====u==n=	0u0011n1un110+1u00u11u11n010101n	================================
11	=nu=============================	=11nu11011un0+000011u0001u=u0uu0	================================
12	====n=========n=n=======u==u====	1011u=01001nuu+0u11010101001=101	================================
13	=======nn=======u===============	=1=0=10==00111+11==1u=1=n0=0=01=	=====n===n==========n==========
14	================================	=u==00===u=011u11==n1=n=1==+====	==u=============================
15	==u=============================	=0=====+00===11=+==01=0=0==+====	================================
16	================================	=0=====+01===n1=+==1==1===0u====	================================
17	================================	==10===nn====1==u===000===11==1=	================================
18	================================	==1====00====1==0==========1====	================================
19	================================	==u====10=======1===10==========	================================
20	================================	==0=============================	================================
21	================================	==1=============================	=====0=uu=====1=n=0=============
22	================================	================================	================================
23	================================	================================	==n=============================
24	================================	================================	================================
25	================================	================================	================================
26	================================	================================	================================
27	================================	================================	================================
28	================================	================================	================================
29	================================	================================	================================
30	================================	================================	================================
31	================================	================================	================================
32	================================	================================	================================
33	================================	================================	================================
34	================================	================================	================================
35	================================	================================	================================

Table 10: Extra conditions influencing the attack for the 36-step SHA-256 differential characteristic

	Conditions	Pro.
A_{16}	$A_{14}[29] = A_{16}[29]$	2^{-1}
A_{17}	$A_{16}[29] = A_{17}[29]$	2^{-1}
E_{16}	$E_{16}[\ 0] \neq E_{16}[13], E_{15}[15] = E_{16}[15], E_{15}[24] = E_{16}[24]$	2^{-3}
E_{17}	$E_{17}[\ 6] \neq E_{17}[19], E_{17}[\ 2] = E_{17}[20]$	2^{-2}
E_{19}	$E_{19}[\ 2] = E_{19}[16]$	2^{-1}
W_5	$W_5[\ 1] \neq W_5[12], W_5[\ 8] \neq W_5[25], W_5[18] \neq W_5[14]$	2^{-3}
W_6	$W_6[\ 0] \neq W_6[28], W_6[\ 9] = W_6[30], W_6[\ 1] \neq W_6[18]$	2^{-3}
W_7	$W_7[\ 1] \neq W_7[12], W_7[\ 8] = W_7[25], W_7[14] \neq W_7[18]$	2^{-3}
W_8	$W_8[14] = W_8[31], W_8[18] = W_8[22], W_8[20] = W_8[31], W_8[\ 9] = W_8[13]$ $W_8[\ 8] = W_8[23], W_8[11] = W_8[22]$	2^{-6}
W_{21}	$W_{21}[31] = W_{21}[\ 1], W_{21}[30] = W_{21}[\ 0], W_{21}[16] = W_{21}[25], W_{21}[14] = W_{21}[21]$	2^{-4}
W_{23}	$W_{23}[\ 4] = W_{23}[\ 6], W_{23}[22] \neq W_{23}[31], W_{23}[20] = W_{23}[27]$	2^{-3}

Table 11: The SFS colliding message pair for 36 steps of SHA-256

CV	414e392b	c207f5e7	56e6b04d	3b8b9fc2	54288441	10492883	8c7b95bb	da1be99e
M	37efdc48	edf7cc76	fd68dc33	5bbf9235	a5ab7e7b	162817f8	6a029e20	9fa012c4
	39887bbf	f14c73e3	9091f022	69f3cd47	aba6b039	e3e7a8e3	a8615f9e	1c664e4c
M'	37efdc48	edf7cc76	fd68dc33	5bbf9235	a5ab7e7b	362817f8	6e429620	bfa012c4
	38896b35	f1447fe3	9091f022	69f3cd47	aba6b039	e7a7a0e3	88615f9e	1c664e4c
hash	514ef8ec	a900d078	478f724c	f35a8d1d	1e5ff01c	03299e4c	d02070dd	5fd1ab65

Table 12: The differential characteristic for 36-step SHA-512

i	∇A_i	∇E_i	∇W_i
-4	==========	==========	
-3	==========	==========	
-2	==========	==========	
-1	==========	==========	
0	==========	==========	==========
1	==========	==========	==========
2	==========	==========	==========
3	==========	=1110==========	==========
4	==========	=0000+1101+++001===1010==========11==========	==========
5	====u==========	=unnn+1110+++110==10100=========1====0=0==0001=1000=+0=1==1=0=1	====u==========0==
6	=====n=====u==u=====n=========n=====u=====u==========	0000nnunnnnnnunn=00nuuu=1===100=000=011=1010nn100111=+010=00=1==	=====u=====u==========1=
7	==========	=10001010n111u1n10un1nn=111=01=00n==0111n10u00u1unnu0un1u=1u=n11	====u==========1==
8	==========	=nuuuu0001010=00==0=11110=0=0+0=10===00=0==110110000=00=0010=0=1	==u=====u=n=====n=====n=====u==========1====
9	==========	=000u1=111=nu1=01=011111111==+0+011=010=1==011110110010=110000==	=u========n==========n==========
10	==========uu===n==========u==u==u====u=n==	11101n001000u01u000100nun0n10nu+n10=0un=1111un0un11u00u1unnn1n=1	==========
11	====u==========	1u10uu010101n10u1000nu0n0n11001n00u1n0100010100nu111001101n1n101	==========
12	==========nun=n=========n=====u=unnn=n==========	00=100110001100111n00uu0111011111=101111101010nnnnnnnnnnnnnnn11	==========
13	=====u=====u=======u==========	011unnnnnnnnnnnnnnnnn1001unn1nn=0=nuuuu0nuuu010000001000111111	=====n=====u==========
14	==========	==110011111110001101110=0uu000n10=u=01n11=01010n10000u0010101n01	====n==========
15	====n==========	===00000000000000001000==00000100=0=00011=0010=0=====0=======1==	==========
16	==========	=====0=====0=====n=1=n==11===0===1===1=====n=1=====1=======1=0	==========
17	==========	====+=u=====u=====0=u=0==========0==========	==========
18	==========	====+=0=====0=====1=0=1==========1==========	==========
19	==========	====n=1=====1=======1==========	==========
20	==========	====0==========	==========
21	==========	====1==========	======n=====n======n==========1==========
22	==========	==========	==========
23	==========	==========	====u==========1==========
24	==========	==========	==========
25	==========	==========	==========
26	==========	==========	==========
27	==========	==========	==========
28	==========	==========	==========
29	==========	==========	==========
30	==========	==========	==========
31	==========	==========	==========
32	==========	==========	==========
33	==========	==========	==========
34	==========	==========	==========
35	==========	==========	==========

Table 13: Extra conditions influencing the attack for the 36-step SHA-512 differential characteristic

	Conditions	Pro.
A_{16}	$A_{14}[59] = A_{16}[59]$	2^{-1}
A_{17}	$A_{15}[59] = A_{16}[59]$	2^{-1}
E_{16}	$E_{16}[55] = E_{16}[59], E_{16}[49] = E_{16}[8]$	2^{-2}
E_{17}	$E_{17}[20] \neq E_{17}[61], E_{17}[16] \neq E_{17}[53], E_{17}[14] = E_{17}[55], E_{17}[10] = E_{17}[47]$ $E_{17}[\ 6] \neq E_{17}[47], E_{17}[\ 2] = E_{17}[39], E_{17}[30] = E_{17}[34], E_{17}[24] \neq E_{17}[28]$ $E_{17}[16] = E_{17}[20]$	2^{-6}
E_{18}	$E_{17}[59] = E_{18}[59]$	2^{-1}
E_{19}	$E_{19}[22] \neq E_{19}[63], E_{19}[55] \neq E_{19}[18], E_{19}[32] = E_{19}[36]$	2^{-3}
W_5	$W_5[53] \neq W_5[60], W_5[52] = W_5[58]$	2^{-2}
W_6	$W_6[52] \neq W_6[59], W_6[45] \neq W_6[52], W_6[44] = W_6[50]$	2^{-2}
W_7	$W_7[53] \neq W_7[60], W_7[52] = W_7[58]$	2^{-2}
W_8	$W_8[55] = W_8[62], W_8[58] = W_8[59], W_8[48] = W_8[55], W_8[47] \neq W_8[53]$ $W_8[46] \neq W_8[53], W_8[38] = W_8[44], W_8[33] \neq W_8[40], W_8[26] \neq W_8[33]$ $W_8[25] = W_8[31]$	2^{-9}
W_{21}	$W_{21}[\ 9] = W_{21}[60], W_{21}[\ 6] \neq W_{21}[48], W_{21}[\ 1] = W_{21}[52], W_{21}[\ 0] = W_{21}[42]$ $W_{21}[35] \neq W_{21}[44], W_{21}[34] = W_{21}[56], W_{21}[29] = W_{21}[38], W_{21}[21] \neq W_{21}[30]$	2^{-8}
W_{25}	$W_{21}[\ 8] = W_{21}[50], W_{25}[37] = W_{25}[46]$	2^{-2}

Table 14: The SFS colliding message pair for 36 steps of SHA-512

CV	f128698fe95be8ac	264034c8f44fe6ed	7b976a39ced01663	7b671a27b6f1da99
	198021f679057203	600969ddc8027d0c	4af8d8a11db1692b	98ba089bc0dee788
M	230a4930d38229e3	2b3c63372261608b	f833951c32f2b6dc	6f4f49447a93b0cc
	f841e91df9c8d69e	a4710d44716286a3	e3310abbec7a9816	66b0241d0bfd564c
	5395fbd4a70df9f3	b0f8774e493b43bc	13287fa0a341d877	f0d19e5ae32f6abf
	f26d943cf8041873	06f2b17bb37eb2b7	2cccf19ec39c66dd	f284ae681c043be9
M'	230a4930d38229e3	2b3c63372261608b	f833951c32f2b6dc	6f4f49447a93b0cc
	f841e91df9c8d69e	ac710d44716286a3	e7390abbec7a9816	6eb0241d0bfd564c
	73c5db55a70df9f3	f0d8764e493b43bc	13287fa0a341d877	f0d19e5ae32f6abf
	f26d943cf8041873	02fab17bb37eb2b7	24ccf19ec39c66dd	f284ae681c043be9
hash	f121c6371d1f7dae	18b855b541505aca	66f9907c6a62d593	f9053b45c041d782
	096ecf2650626b26	fa00b8af66c6af89	d3c534d9fe0f249d	c2f16b3fcbc3032d

Table 15: The differential characteristic for 37-step SHA-256

i	∇A_i	∇E_i	∇W_i
-4	================================	================================	
-3	================================	================================	
-2	================================	================================	
-1	================================	================================	
0	================================	================================	================================
1	================================	================================	================================
2	================================	================================	================================
3	================================	================================	================================
4	================================	000=============================	================================
5	================================	111=0=====1==0=====01===++01====	================================
6	=nu=============================	uuu=1011=00=10111=0111=0++1011==	==n=============================
7	==========n====n====n======n====	10n=u01001n00n00010nu110unuu0001	=====u===u==========n==========
8	================================	011=1n+1=n11u1101=001u1u1010=u=1	==u=============================
9	================================	1=0110+=001=1100=+110100111u=0=+	=====u=u=======n===u==n=u=n=u=n=
10	======u==========u==============	10n001u110101==01+u1101011=1110+	============n======u=n==========
11	====u=====u=========u=n====u==n=	=01un000011101=n0n0u1uu101011u1u	================================
12	=nu=============================	01101uuuuunu010110110n1u0u0uu0n0	================================
13	====u=========u=u=======n==u====	0+10n110111unn+0n101110110001001	================================
14	======u=n=======n===============	=+0=1000000111+=1==1n010u0=0101=	=0===u===n=1=1====1=u=1=====1=1=
15	================================	=u==01===n=011n01==u0=u=1==+====	==u=============================
16	==u=============================	=0=====+00===11=+==01=0=0==+====	================================
17	================================	=0=====+01===u1=+==0==1===1u====	================================
18	================================	==10===uu====0==n===111===01==1=	================================
19	================================	==1====00====1==0==========1====	================================
20	================================	==u====10=======1===00==========	================================
21	================================	==0=============================	================================
22	================================	==1=============================	=====0=nn=====1=u=1============
23	================================	================================	================================
24	================================	================================	==n=============================
25	================================	================================	================================
26	================================	================================	================================
27	================================	================================	================================
28	================================	================================	================================
29	================================	================================	================================
30	================================	================================	================================
31	================================	================================	================================
32	================================	================================	================================
33	================================	================================	================================
34	================================	================================	================================
35	================================	================================	================================
36	================================	================================	================================

Table 16: Extra conditions influencing the attack for the 37-step SHA-256 differential characteristic

	Conditions	Pro.
A_{16}	$A_{15}[15] \neq A_{16}[15], A_{15}[23] = A_{16}[23], A_{15}[25] = A_{16}[25], A_{16}[\ 9] \neq A_{16}[20]$ $A_{16}[18] \neq A_{16}[\ 6], A_{16}[\ 8] = A_{16}[17]$	2^{-6}
A_{17}	$A_{16}[29] = A_{17}[29]$	2^{-1}
A_{18}	$A_{17}[29] = A_{18}[29]$	2^{-1}
E_{16}	$E_{15}[\ 4] = E_{16}[\ 4]$	2^{-1}
E_{17}	$E_{17}[\ 0] \neq E_{17}[13], E_{16}[24] = E_{17}[24], E_{16}[15] = E_{17}[15]$	2^{-3}
E_{18}	$E_{18}[\ 6] \neq E_{18}[19], E_{18}[\ 2] = E_{18}[20]$	2^{-2}
E_{20}	$E_{20}[\ 2] \neq E_{20}[16]$	2^{-1}
W_6	$W_6[\ 1] = W_6[12], W_6[\ 8] \neq W_6[25], W_6[14] = W_6[18]$	2^{-3}
W_7	$W_7[\ 0] = W_7[28], W_7[\ 9] = W_7[30], W_7[\ 1] = W_7[18]$	2^{-3}
W_8	$W_8[\ 1] \neq W_8[12], W_8[\ 8] \neq W_8[25], W_8[14] = W_8[18]$	2^{-3}
W_{22}	$W_{22}[31] \neq W_{22}[\ 1], W_{22}[30] \neq W_{22}[\ 0], W_{22}[16] \neq W_{22}[25], W_{22}[14] \neq W_{22}[21]$	2^{-4}
W_{24}	$W_{24}[\ 4] \neq W_{24}[\ 6], W_{24}[22] = W_{24}[31], W_{24}[20] \neq W_{24}[27]$	2^{-3}

Table 17: The SFS colliding message pair for 37 steps of SHA-256

CV	63b4986c	35d83dc0	c98894e4	784e08fc	78a7f752	5ed877a8	315a2db3	d5614eb4
M	4d7f86ff	32ece589	d8accfc4	2cc2e433	068b5b34	eba68a28	2fe12cad	6aa26b0c
	9f0d78e1	681b8277	faa9c7e0	56aed439	cc2dbbc2	dd2ba0fc	b95d377b	5dd43a81
M'	4d7f86ff	32ece589	d8accfc4	2cc2e433	068b5b34	eba68a28	0fe12cad	6ee2630c
	bf0d78e1	6d1a90dd	faa1d3e0	56aed439	cc2dbbc2	dd2ba0fc	bd1d3f7b	7dd43a81
hash	a856d46e	4b46eb28	4935248c	92a2fc98	e0fb2610	10a9951f	54264f5b	80954580

Table 18: The differential characteristic for 37-step SHA-512

i	∇A_i	∇E_i	∇W_i
-4	==	==	
-3	==	==	
-2	==	==	
-1	==	==	
0	==	==	==
1	==	==	==
2	==	==	==
3	==	==	==
4	==	=++==	==
5	==	=+++1111++0110+++=+10==================11==================	==
6	==u===	=un+0110++1001+++++11=========0====1=1==+11======0===+1===1=0==	==u==0====
7	===n=====u=====n===u========n====u======n===================	=01nunnnnnnnuuuuu+unn+1+1+011+001=001=010+uu======1===+0===0=0=0	===n=====u==1====
8	==	=001n10u110n1n101n1uu+1+1+011+0n0110n0n11nuu11====u11=uu01=u=1==	==u==0====
9	==	==0u1101100=11110100unnnnnnnnn0010nu=1111=10100101101100110011	n======u=n======u=====n======n==================================
10	==	==u11=01100100=un00100010111111111nuuu1110011101=1111110110101001	========u==========n===u
11	================u===uu=u===u================u=-n==n===u=u====	010n000100u0110111nnn100nn1011uu10010u10011=n01u0nu=nu011ununn0001	==
12	==unu==	1001u0110nu00u0010001uu1u0001n11n1n1110110=0un001011nuu10010=1un	==
13	================unu=u========n====u=unnn=n===================	unnnnnnnnnnnnnnnnn0nnnn1n001+n0nn0nuu1u01n11101001=1000111nu1111	==
14	====n=====u=======n==	110011110=00001100=110111=0=+=u01=110=n==01=11=====1011=-011=111	===n======u==
15	==	011000000001101111111101nn==1u010n=10n=1=10=+=n===1=n======0u01==	==n==
16	==n==	====+======+=====1=1=0=-=11===1=1=1===0=0====+=0=====0======1====	==
17	==	====+======+=====u=0=u=-=11===1=-=1===1======n=1=====1======1=1==	==
18	==	==+=n=====u=====0=n=0=========================0==================	==
19	==	==+=0=====0=====1=0=1=====================1==================	==
20	==	==n=1=====1=======1==	==
21	==	==0==	==
22	==	==1==	===u=====n=======u=========================0==================
23	==	==	==
24	==	==	==u==
25	==	==	==
26	==	==	==
27	==	==	==
28	==	==	==
29	==	==	==
30	==	==	==
31	==	==	==
32	==	==	==
33	==	==	==
34	==	==	==
35	==	==	==

Table 19: Extra conditions influencing the attack for the 37-step SHA-512 differential characteristic

	Conditions	Pro.
A_{16}	$A_{15}[45] \neq A_{16}[45], A_{15}[53] = A_{16}[53], A_{15}[59] = A_{16}[59], A_{16}[50] \neq A_{16}[56]$ $A_{16}[55] \neq A_{16}[\ 2], A_{16}[\ 3] = A_{16}[\ 8]$	2^{-6}
A_{17}	$A_{16}[61] = A_{17}[61]$	2^{-1}
A_{18}	$A_{17}[61] = A_{18}[61]$	2^{-1}
E_{16}	$E_{15}[20] = E_{16}[20]$	2^{-1}
E_{17}	$E_{17}[57] \neq E_{17}[61], E_{17}[10] \neq E_{17}[51], E_{16}[59] = E_{17}[59], E_{16}[53] = E_{17}[53]$ $E_{18}[22] = E_{18}[63], E_{18}[18] \neq E_{18}[55], E_{18}[16] \neq E_{18}[57], E_{18}[12] \neq E_{18}[49]$	2^{-4}
E_{18}	$E_{18}[\ 8] \neq E_{18}[49], E_{18}[\ 4] \neq E_{18}[41], E_{18}[32] \neq E_{18}[36], E_{18}[26] = E_{18}[30]$ $E_{18}[18] = E_{18}[22]$	2^{-9}
E_{19}	$E_{18}[61] = E_{19}[61]$	2^{-1}
E_{20}	$E_{20}[22] = E_{20}[63], E_{20}[18] = E_{20}[55], E_{20}[32] \neq E_{20}[36]$	2^{-3}
W_6	$W_6[55] \neq W_6[62], W_6[54] = W_6[60]$	2^{-2}
W_7	$W_7[54] = W_7[61], W_7[47] = W_7[54], W_7[46] \neq W_7[52]$	2^{-3}
W_8	$W_8[55] \neq W_8[62], W_8[54] \neq W_8[60]$	2^{-2}
W_{22}	$W_{22}[11] = W_{22}[62], W_{22}[\ 8] = W_{22}[50], W_{22}[\ 3] = W_{22}[54], W_{22}[\ 2] \neq W_{22}[44]$ $W_{22}[37] = W_{22}[46], W_{22}[36] \neq W_{22}[58], W_{22}[31] = W_{22}[40], W_{22}[23] = W_{22}[32]$	2^{-8}
W_{24}	$W_{24}[10] \neq W_{24}[52], W_{24}[39] = W_{24}[48], W_{24}[\ 6] \neq W_{24}[19]$	2^{-3}

Table 20: The SFS colliding message pair for 37 steps of SHA-512

CV	e13f762148155018	257ea2102aa5cc49	ac3a1c4c18358111	49c346ed04ebb556
	3ce92b731dc59e88	1c89208791930dc1	f322c6752f284e63	8b2de269a28efa96
M	9e6f8fc6be06d6a2	e8f5c62dd9426c65	887815c2aafc95a0	b3c42bfc03936743
	82ee7e0a1a3425e1	50906d2509e692cd	089787e261268687	bccdc39ccb02c6be
	5f5d397f451e81cd	84c727e603079d93	ec1edc25805f575a	a84fa4027bf5a716
	c20febc2034bb057	575aa92b223833b2	be17c10c936e0e36	a04d0b5a2e66e190
M'	9e6f8fc6be06d6a2	e8f5c62dd9426c65	887815c2aafc95a0	b3c42bfc03936743
	82ee7e0a1a3425e1	50906d2509e692cd	289787e261268687	acedc39ccb02c6be
	7f5d397f451e81cd	0587a5e203079d93	ec9ed825805f575b	a84fa4027bf5a716
	c20febc2034bb057	575aa92b223833b2	ae37c10c936e0e36	804d0b5a2e66e190
hash	e28c6359c9f80b41	a9b888197e8518db	54ad6b94458dbe4e	8c2d66ee1e543e03
	d6f7cd9f0c2d3a28	a6db911fae70158d	dbadea72fe5ab84d	9c63cd2c39cbee6c

Table 21: The differential characteristic for 39-step SHA-256

i	∇A_i	∇E_i	∇W_i
-4	================================	================================	
-3	================================	================================	
-2	================================	================================	
-1	================================	================================	
0	================================	================================	================================
1	================================	================================	================================
2	================================	================================	================================
3	================================	================================	================================
4	================================	================================	================================
5	================================	================================	================================
6	================================	================================	================================
7	================================	====1===0=100======0======0=====	================================
8	=n==============================	=n0=0=001=011=11=0=100+=001=1=0=	=n==============================
9	====n===n===u=n====n===u==u=====	=010u100n=unn==1000n11+001u=0111	====u===u==========n===========
10	================================	u0n1111n00u110110u10uuu1uuu0u0n1	=u==============================
11	================================	1u1=u1110000n00011=1100n001=0=01	=====nn=======n===u===uu==uu=n==
12	=====n==nn======================	1n111u00nu0010u00011100110010=0=	==========u=======uu==========
13	==u=u==u=u==u==nn====u======n=uu	10u0u0n001uuuu01u01uu=nu0001u001	================================
14	=n==========u==========n=====	=001110=1100101=0==00=01====0===	================================
15	===n=========uu========u==n=====	==0u0=1=1=0000==11=01100====0=11	================================
16	=====u=n=======u===============	===1====0=======1==001=n======0	====u===u=1=0====1=n=0=====1=1==
17	================================	n==1====n=====u=u==uuu=1======1n	=u==============================
18	=u==============================	0======00=====100==000=0=======0	================================
19	================================	0======01===u0111==011===1n====1	================================
20	================================	=10===uu====1==u===011===11==0==	================================
21	================================	=1====00====1==0==========1=====	================================
22	================================	=u====10=======1===01==========	================================
23	================================	=0==============================	================================
24	================================	=1==============================	====1=nn=====1=n=1==============
25	================================	================================	================================
26	================================	================================	=n==============================
27	================================	================================	================================
28	================================	================================	================================
29	================================	================================	================================
30	================================	================================	================================
31	================================	================================	================================
32	================================	================================	================================
33	================================	================================	================================
34	================================	================================	================================
35	================================	================================	================================
36	================================	================================	================================
37	================================	================================	================================
38	================================	================================	================================

Table 22: Extra conditions influencing the attack for the 39-step SHA-256 differential characteristic

	Conditions	Pro.
A_{16}	$A_{14}[\ 8] = A_{16}[\ 8], A_{14}[17] = A_{16}[17], A_{14}[28] \neq A_{16}[28], A_{15}[30] \neq A_{16}[30]$ $A_{16}[\ 4] \neq A_{16}[15], A_{16}[\ 5] = A_{16}[25], A_{16}[\ 6] = A_{16}[17], A_{16}[\ 1] \neq A_{16}[13]$ $A_{16}[\ 3] = A_{16}[15], A_{16}[\ 4] \neq A_{16}[27], A_{16}[\ 3] = A_{16}[12], A_{16}[\ 5] \neq A_{16}[14]$ $A_{16}[\ 7] \neq A_{16}[28]$	2^{-13}
A_{17}	$A_{16}[\ 5] \neq A_{17}[\ 5], A_{16}[\ 8] = A_{17}[\ 8], A_{15}[16] = A_{17}[16], A_{16}[17] = A_{17}[17]$ $A_{16}[18] \neq A_{17}[18], A_{15}[24] = A_{17}[24], A_{15}[26] = A_{17}[26], A_{16}[28] = A_{17}[28]$	2^{-8}
A_{18}	$A_{17}[16] = A_{18}[16], A_{17}[24] = A_{18}[24], A_{17}[26] = A_{18}[26], A_{17}[30] = A_{18}[30]$ $A_{18}[10] = A_{17}[21], A_{18}[\ 7] \neq A_{18}[19], A_{18}[\ 9] \neq A_{18}[18]$	2^{-7}
A_{19}	$A_{17}[30] = A_{19}[30]$	2^{-1}
A_{20}	$A_{19}[30] = A_{20}[30]$	2^{-1}
E_{16}	$E_{15}[17] = E_{16}[17], E_{15}[31] = E_{16}[31], E_{16}[3] = E_{16}[22], E_{16}[21] = E_{16}[26]$ $E_{16}[13] = E_{16}[27]$	2^{-5}
E_{17}	$E_{17}[\ 5] = E_{17}[24], E_{17}[\ 4] = E_{17}[\ 9], E_{17}[\ 5] = E_{17}[19], E_{17}[\ 4] \neq E_{17}[18]$ $E_{17}[\ 3] = E_{17}[30], E_{17}[14] \neq E_{17}[27], E_{17}[13] \neq E_{17}[26], E_{17}[25] = E_{17}[30]$ $E_{17}[24] = E_{17}[29], E_{17}[\ 5] \neq E_{17}[18], E_{17}[\ 4] \neq E_{17}[22], E_{17}[\ 2] = E_{17}[20]$ $E_{17}[13] = E_{17}[18], E_{17}[16] \neq E_{17}[30], E_{17}[\ 7] \neq E_{17}[26], E_{17}[\ 6] = E_{17}[25]$	2^{-16}
E_{18}	$E_{17}[\ 5] \neq E_{18}[\ 5], E_{17}[19] = E_{18}[19]$	2^{-2}
E_{19}	$E_{19}[\ 1] = E_{19}[14], E_{18}[25] = E_{19}[25]$	2^{-2}
E_{20}	$E_{20}[\ 7] = E_{20}[20], E_{20}[\ 3] = E_{21}[20]$	2^{-2}
E_{21}	$E_{21}[\ 3] \neq E_{21}[17]$	2^{-1}
W_{16}	$W_{16}[\ 1] \neq W_{16}[29], W_{16}[10] \neq W_{16}[31], W_{16}[\ 0] \neq W_{16}[30], W_{16}[18] = W_{16}[25]$ $W_{16}[20] \neq W_{16}[29], W_{16}[16] \neq W_{16}[25]$	2^{-6}
W_{17}	$W_{17}[\ 2] \neq W_{17}[13], W_{17}[\ 9] \neq W_{17}[26], W_{17}[15] = W_{17}[19], W_{17}[\ 5] \neq W_{17}[\ 7]$ $W_{17}[\ 0] \neq W_{17}[23], W_{17}[21] = W_{17}[28]$	2^{-6}
W_{23}	$W_{23}[\ 2] \neq W_{23}[\ 0], W_{23}[31] \neq W_{23}[\ 1], W_{23}[17] = W_{23}[26], W_{23}[15] \neq W_{23}[22]$	2^{-4}
W_{25}	$W_{25}[\ 5] = W_{25}[\ 7], W_{25}[21] \neq W_{25}[28], W_{25}[\ 0] = W_{25}[23]$	2^{-3}

Table 23: The SFS colliding message pair for 39 steps of SHA-256

CV	b21e3e63	d12deef2	71a28861	1c40a0d4	15bb07fb	b326f54c	020b80dd	0e6446c1
M	5ee50e16	b5a4f02b	a58b289e	f713de5d	3b96d2db	215da7e2	3674669a	2ca65d8d
	e99c40e3	172e9a20	02e88746	07021c8f	e8c6e622	b06304c1	43660040	4891771f
M'	5ee50e16	b5a4f02b	a58b289e	f713de5d	3b96d2db	215da7e2	3674669a	2ca65d8d
	a99c40e3	1fae8a20	42e88746	01003fbb	e8d6fe22	b06304c1	43660040	4891771f
hash	f42bbd82	0f057a39	267fcb51	e202bc25	81e4ed41	d74409dd	3247beb6	ec7ed559

Table 24: The differential characteristic for 39-step SHA-512

i	∇A_i	∇E_i	∇W_i
-4	==========	==========	
-3	==========	==========	
-2	==========	==========	
-1	==========	==========	
0	==========	==========	==========
1	==========	==========	==========
2	==========	==========	==========
3	==========	==========	==========
4	==========	==========	==========
5	==========	==========	==========
6	==========	1==========	==========
7	==========	0+++0+++++00++0=0+0==========00==========	==========
8	n==========	u+++0+++++01++0=0+1===00110101=001=10===11====+=+++1=0=+1+======	u==========
9	=u======n=====n===n=========n=====u====n=n==========	nunuuuuuunuuuuu=nuu==0100110001000000110nu110=+=+++0=1=+0+10101=	=u======n==========
10	=====n==========n==========	u010001111un00n1n10010unnnnnnn1nuu0un101nu0111n1uunn0n0unn010101	n==========
11	========n=====n===n==n==========	unuu1u010010101101111n110000001000n100un1uunn110111101011011 0u00	=====n=u======n=====n======u==========nu=
12	==========n==========u===n===u===u==	1u001u001uuu1un001u100110101010100100nn0nn101=100001u111nnuuuu10	======u==========n==========n==
13	nnn==========nnu==nun========u===uu=nu==n==n=====nn======	n0n0u1nnnnnnn1nnnnnnnnnnnnnnnn0nuu00110unuuu=uu=n1=nuuu01101001	==========
14	u==========u==========	u01=1001011111=1111110000000010=001=010101111=0010==u000100000==	==========
15	==========n=unn========nu=====u=unnnnu==========	0=1=1=0000011=10010111111111011=11101==n10111=11=0==0111=1====1=	==========
16	==u=====u=======n==========	1==========0=1===0=u=1===1=n==1===1=====0==1====0======	=u======n==========1======0======
17	==========	==========+=1=0=n=n===n=0=u===n=1==1=+=u=====n=======u=1====	n==========0======
18	n==========	==+=====+=====+=1=1=0=0===0=0=0===1=1====+=0=====0=======0======	==========
19	==========	==+=====+=====n=0=u=1=1===1===1===1======n=1=====1=======1=1====	==========
20	==========	+=u=====u=====0=n=0==========0==========	==========
21	==========	+=0=====0=====1=0=1==========1==========	==========
22	==========	u=1=====1=======1==========	==========
23	==========	0==========	==========
24	==========	1==========	==n=====n=======u==========0==========
25	==========	==========	==========
26	==========	==========	n==========
27	==========	==========	==========
28	==========	==========	==========
29	==========	==========	==========
30	==========	==========	==========
31	==========	==========	==========
32	==========	==========	==========
33	==========	==========	==========
34	==========	==========	==========
35	==========	==========	==========
36	==========	==========	==========
37	==========	==========	==========
38	==========	==========	==========

Table 25: Extra conditions influencing the attack for the 39-step SHA-512 differential characteristic

	Conditions	Pro.
A_{16}	$A_{14}[22] \neq A_{16}[22], A_{14}[23] = A_{16}[23], A_{14}[24] = A_{16}[24], A_{14}[26] = A_{16}[26]$ $A_{14}[27] \neq A_{16}[27], A_{14}[29] \neq A_{16}[29], A_{14}[35] = A_{16}[35], A_{14}[36] \neq A_{16}[36]$ $A_{14}[45] = A_{16}[45], A_{14}[46] = A_{16}[46], A_{14}[49] = A_{16}[49], A_{14}[63] = A_{16}[63]$ $A_{16}[36] = A_{16}[42], A_{16}[41] \neq A_{16}[52], A_{16}[44] \neq A_{16}[50], A_{16}[53] \neq A_{16}[58]$ $A_{16}[49] = A_{16}[60], A_{16}[50] = A_{16}[56], A_{16}[\ 3] \neq A_{16}[\ 8]$	2^{-19}
A_{17}	$A_{16}[22] = A_{17}[22], A_{16}[23] = A_{17}[23], A_{16}[24] = A_{17}[24], A_{16}[25] \neq A_{17}[25]$ $A_{16}[26] \neq A_{17}[26], A_{16}[27] \neq A_{17}[27], A_{16}[29] = A_{17}[29], A_{16}[35] = A_{17}[35]$ $A_{16}[36] = A_{17}[36], A_{16}[45] = A_{17}[45], A_{16}[46] = A_{17}[46], A_{16}[49] = A_{17}[49]$ $A_{15}[55] = A_{17}[55], A_{15}[61] = A_{17}[61], A_{16}[63] = A_{17}[63]$	2^{-15}
A_{18}	$A_{17}[47] = A_{18}[47], A_{17}[55] = A_{18}[55], A_{17}[61] = A_{18}[61], A_{18}[52] \neq A_{18}[58]$ $A_{18}[57] \neq A_{17}[\ 4], A_{18}[\ 5] \neq A_{18}[10]$	2^{-6}
A_{19}	$A_{17}[63] = A_{19}[63]$	2^{-1}
A_{20}	$A_{19}[63] = A_{20}[63]$	2^{-1}
E_{16}	$E_{16}[\ 8] \neq E_{16}[12], E_{16}[\ 0] = E_{16}[\ 4], E_{16}[39] \neq E_{16}[62], E_{16}[31] \neq E_{16}[58]$ $E_{16}[31] \neq E_{16}[54], E_{16}[23] \neq E_{16}[50]$	2^{-6}
E_{17}	$E_{17}[57] \neq E_{17}[61], E_{17}[51] = E_{17}[55], E_{17}[\ 2] = E_{17}[39], E_{17}[25] = E_{17}[52]$	2^{-4}
E_{18}	$E_{17}[22] \neq E_{18}[22], E_{17}[49] = E_{18}[49]$	2^{-2}
E_{19}	$E_{19}[59] = E_{19}[63], E_{19}[12] = E_{19}[53], E_{18}[61] = E_{19}[61], E_{18}[55] = E_{19}[55]$	2^{-4}
E_{20}	$E_{20}[\ 1] \neq E_{20}[24], E_{20}[20] = E_{20}[57], E_{20}[18] = E_{21}[59], E_{20}[14] = E_{20}[51]$ $E_{20}[10] = E_{20}[51], E_{20}[\ 6] \neq E_{20}[43], E_{20}[34] \neq E_{20}[38], E_{20}[28] = E_{20}[32]$ $E_{20}[20] = E_{20}[24]$	2^{-9}
E_{21}	$E_{20}[63] = E_{21}[63]$	2^{-1}
E_{22}	$E_{22}[\ 3] = E_{22}[26], E_{22}[22] \neq E_{22}[59], E_{22}[36] = E_{22}[40]$	2^{-3}
W_{16}	$W_{16}[11] \neq W_{16}[53], W_{16}[\ 4] \neq W_{16}[46], W_{16}[40] = W_{16}[49], W_{16}[33] = W_{16}[42]$ $W_{16}[\ 7] = W_{16}[20], W_{16}[56] = W_{16}[63], W_{16}[49] = W_{16}[56], W_{16}[48] \neq W_{16}[54]$	2^{-8}
W_{17}	$W_{17}[12] = W_{17}[54], W_{17}[41] \neq W_{17}[50], W_{17}[\ 8] \neq W_{17}[21], W_{17}[57] \neq W_{17}[\ 0]$ $W_{17}[56] = W_{17}[62]$	2^{-5}
W_{23}	$W_{23}[10] \neq W_{23}[52], W_{23}[\ 5] \neq W_{23}[56], W_{23}[\ 4] = W_{23}[46], W_{23}[39] = W_{23}[48]$ $W_{23}[33] \neq W_{23}[42], W_{23}[25] \neq W_{23}[34]$	2^{-6}
W_{25}	$W_{25}[12] = W_{25}[54], W_{25}[41] \neq W_{25}[50], W_{25}[\ 8] = W_{25}[21]$	2^{-3}

Table 26: The SFS colliding message pair for 39 steps of SHA-512

CV	e6ddec29a1f2e312	e9a2c38135d94804	3ae0345d8668c89a	d80ad161859b123b
	386b9076edf4e410	fe173ab1b50b6d9d	179926c6b9b05d01	cc507d00f33268d1
M	c4900297c267d68b	d4d1795ea32cb267	f73b9b24f6d36452	3201f73c35ebd0e
	3ca7b3d631444403	c4b2b47463e50ce0	e66b0df1f6c004cf	4eeecf2ee6968cef
	176574fcdf766ce9	80d3b6b153db847d	80d7254e8e3e8011	4ce2df67fe5f8fec
	c885ba5bf5403d6c	e02bb1fcdcd75830	fba13633dc5703e6	bf9503d51a5963de
M'	c4900297c267d68b	d4d1795ea32cb267	f73b9b24f6d36452	03201f73c35ebd0e
	3ca7b3d631444403	c4b2b47463e50ce0	e66b0df1f6c004cf	4eeecf2ee6968cef
	976574fcdf766ce9	c053b6b153db847d	00d7254e8e3e8011	49e0d777fe5f8fea
	ca85aa5bf5403d68	e02bb1fcdcd75830	fba13633dc5703e6	bf9503d51a5963de
hash	6c24b4bce61ec0e6	8113d6fcc5b6476f	6b16f18bfe633655	15110d4e9bc0e71a
	7971de3ef60154c3	f4743b50b393d3ea	08b96857f2d6d2af	65cdce3a9363bccb