# More Insights on Blockcipher-Based Hash Functions

Yiyuan Luo, Xuejia Lai

Department of Computer Science and Engineering, Shanghai Jiaotong University
luoyiyuan@sjtu.edu.cn

**Abstract.** In this paper we give more insights on the security of blockcipher-based hash functions. We give a very simple criterion to build a secure large class of Single-Block-Length (SBL) or double call Double-Block-Length (DBL) compression functions based on $(kn, n)$ blockciphers, where $kn$ is the key length and $n$ is the block length and $k$ is an integer.

This criterion is simpler than previous works in the literature. Based on the criterion, we can get many results from this criterion, and we can get a conclusion on such class of blockcipher-based hash functions. We solved the open problem left by Hirose. Our results show that to build a secure double call DBL compression function, it is required $k >= m + 1$ where $m$ is the number of message blocks. Thus, we can only build rate 1/2 secure double DBL blockcipher-based compression functions if $k == 2$.

At last, we pointed out flaws in Stam's theorem about supercharged functions and gave a revision of this theorem and added another condition for the security of supercharged compression functions.

## 1 Introduction

Cryptographic hash function, which is defined as an admissible algorithm that uniformly maps arbitrary length inputs to fixed length outputs, is widely used as a pivotal primitive for ensuring the integrity of information. A hash function usually consists of iteration of a compression function with fixed input and output length. One first design a fixed domain compression function and then extend the domain to an arbitrary domain by iterating the compression function several times.

As flaws in popular classic hash functions MD5 [25] and SHA-1 [3] have been attacked [30,29], NIST has launched the competition for a new hash function standard SHA-3. Actually, the popular design of hash functions are from the idea design of block ciphers, either explicitly such as MDC-2 [12] and other schemes [19] or implicity such as MD5. In the five finalists of the SHA-3 competition, 2 of them (BLAKE, Skein) are blockcipher-based design, the other three are permutation-based design, which are related to blockciphers [22]. Thus, hash functions composed of blockciphers are very important and worth of study.

Currently blockcipher-based hash functions are classified into single block length (SBL) hash functions and double block length (DBL) hash functions. For

SBL hash functions, the length of output is equal to that of the blockcipher, while for DBL hash functions, the length of the output is twice larger than that of the underlined blockcipher. For a typical blockcipher such as AES, the block length is 128 bits, thus a hash function with 128-bit output is no longer secure against the birthday attack. Thus, more and more works start to focus on blockcipher-based functions with larger output of length [5,6,7,8,9,10,11,13,14,15,17,20,21,23,26,28].

In [24], Preneel, Govaerts, and Vandewalle (PGV) systematically study SBL blockcipher-based hash functions. They regarded 12 out of 64 PGV schemes as secure. They focused on attacks, not on proofs. Later in [1], Black *et al.* systematically studied SBL blockcipher-based hash functions in the ideal cipher model. They proved that the 12 schemes considered by PGV really are secure, furthermore, they found an additional 8 of the PGV schemes are just as collision resistant in the iteration. After that, to prove the collision resistance or preimage resistance of a blockcipher-based hash function in the ideal model became the standard way of research.

In [27], Stam considered generalizations of PGV functions gave a conclusive discussion on SBL blockcipher-based functions. Stam proposed criterions for a secure blockcipher-based compressions and secure blockcipher-based functions in the iteration, though the compression function is not that secure. Stam also studied chopped, overloaded and supercharged compression functions.

For double call DBL hash functions, Knudsen *et al.* [13] discussed the security of DBL hash functions with rate 1 based on $(n, n)$ blockciphers. Hohl *et al.* [11] discussed the security of compression functions of DBL hash functions with rate 1/2. Satoh *et al.* [26] and Hattori *et al.* [6] and Hirose [9,10] discussed DBL hash functions with rate 1 based on $(n, 2n)$ blockciphers. Fleischmann *et al.* [5,4] address the collision resistance of two old DBL constructions known as Abreast-DM and Tandem-DM [15,14].

Özen and Stam [23] proposed a novel framework for DBL blockcipher-based hash functions. For single call DBL blockcipher-based hash functions, Lucks [17] first proposed a collision resistant single call DBL blockcipher-based hash function in the iteration. Later, Stam [27] proposed a single call rate-1 DBL blockcipher-based supercharged compression that is opimally collision resistant up to a logarithmic factor.

In this paper we give more insights on the security of blockcipher-based hash functions. We give a very simple criterion to build a large class of secure SBL or DBL compression functions based on $(kn, n)$ blockciphers. We gave a new definition (security rate) for blockcipher-based hash functions. This new definition is different from the efficiency rate which only measures the efficiency of a blockcipher-based hash functions.

This criterion is simpler than previous works in the literature. We classified such a large class of blockcipher-hash functions into **Type-1, Type-2, Type-3, Type-4** compression functions. Based on this criterion, we counted the exact number of optimum security .

There are 12 **Type-1**, 672 **Type-2**, 7676928 **Type-3** optimum secure compression functions and we proved their security in the ideal cipher model. We

pointed out that Abreast-DM, Tandem-DM and Hirose's Schemes are just special cases of **Type-3** compression functions. Thus they are optimum secure (collision resistance and preimage resistance) in the ideal cipher model.

Based on the criterion, there doesn't exist any **Type-4** optimum secure compression functions. We also found collision attacks on **Type-4** hash functions in the iteration with complexity much lower than the birthday complexity. Thus we solved the open problem left by Hirose that if there exists optimum collision resistant **Type-4** hash functions in the iteration.

Our results show that to build a secure double call DBL compression function, it is required $k >= m+1$ where $m$ is the number of message blocks. Thus, we can only build rate 1/2 secure double DBL blockcipher-based compression functions if $k == 2$.

At last, we pointed out flaws in Stam's theorem about supercharged functions and gave a revision of this theorem and added another condition for the security of supercharged compression functions.

Through our analysis, it seems optimum collision resistance and (2nd) preimage resistance Rate-1 DBL compression functions doesn't exist.

## 2 Preliminaries

**Lemma 1.** *[16] The number of $m \times r$ matrices of rank $r$ over $GF(2)$ is:*

$$N(m,r) = 2^{r(r-1)/2} \prod_{i=0}^{r-1} (2^{m-i} - 1).$$

**Definition 1.** *A compression function $F$ is optimum secure if it is optimum against (second) preimage attack and collision attack.*

**Definition 2.** *Let $F$ be a compression function composed of block ciphers, $m$ is number of message blocks in terms of the block length of the underlined blockcipher, $N$ is the number of cipher calls in $F$, the efficiency rate $r$ defined below is an index of efficiency:*

$$r = \frac{m}{N}.$$

The original definition of hash rate is in [13]. We realized that this definition only related to the efficiency of hash. It has no relationship with the key length of the underlined blockcipher. We can modify it to a more accurate definition we called security rate:

**Definition 3.** *Let $F$ be a compression function composed of blockciphers, $m$ is the number of message blocks in terms of the block length of the underlined blockcipher, $N$ is the number of cipher calls in $F$, $K$ is the key length of the blockcipher and $L$ is the output length of $F$, the security rate $R$ defined below is an index of security:*

$$R = \frac{m \cdot L}{N \cdot K}.$$

The security rate of a compression function $F$ can be seemed as an index of the security of compression function. Its security related to the input and output length of $F$, the key length of the underlined blockciphers and the number of cipher calls. This definition is more generic than the efficiency rate. The security rate of a classical Davies-Meyer compression function [24] based on a $(n, n)$ blockcipher is 1, and the security rate will still be 1 even it is based on a $(2n, n)$ blockcipher. This definition can also be applied to DBL block-cipher-based hash functions and thus reduces the complexity of categories of block-cipher-based hash functions.

If $F$ only makes one call to the underlined blockcipher, then it is a single-call compression function. If $F$ makes two calls to the underlined blockcipher, it is a double-call compression function. In the following we classify the block-cipher-based compression function into different types.

**Definition 4.** *Types of Block-Cipher-Based Compression Functions:*

- ***Type-1:*** *For a single-call Rate-1 SBL compression function based on a $(n, n)$ blockcipher $E$, we use the PGV model [24]. The compression function $f(H_{i-1}, M_i) = E_A(B) \oplus C$, where $(A, B, C)$ is a linear composition of $\{H_{i-1}, M_i\}$. Namely,*

$$\begin{pmatrix} A \\ B \\ C \end{pmatrix} = L_1 \begin{pmatrix} H_{i-1} \\ M_i \end{pmatrix}$$

  *and $L_1$ is a $3 \times 2$ $\{0,1\}$-matrix. Here we neglect the constant in the PGV model since it is not related to the security.*
- ***Type-2:*** *For a single-call Rate-1 SBL compression function based on a $(2n, n)$ blockcipher $E$, we write the compression function $f$ as: $f(H_{i-1}, M_i^1, M_i^2) = E_{A\|B}(C) \oplus D$, where*

$$\begin{pmatrix} A \\ B \\ C \\ D \end{pmatrix} = L_2 \begin{pmatrix} H_{i-1} \\ M_i^1 \\ M_i^2 \end{pmatrix}$$

  *and $L_2$ is a $4 \times 3$ $\{0,1\}$-matrix.*
- ***Type-3:*** *For a double-call Rate-$\frac{1}{2}$ DBL compression function based on two independent $(2n, n)$ blockciphers $E^U$ and $E^L$, we write the compression function $(H_i, G_i) = F(H_{i-1}, G_{i-1}, M_i)$ as:*

$$\begin{cases} H_i = f(H_{i-1}, G_{i-1}, M_i) = E^U_{A\|B}(C) \oplus D \\ G_i = g(H_{i-1}, G_{i-1}, M_i) = E^L_{W\|X}(Y) \oplus Z \end{cases}$$

  *where*

$$\begin{pmatrix} A \\ B \\ C \\ D \end{pmatrix} = U \begin{pmatrix} H_{i-1} \\ G_{i-1} \\ M_i \end{pmatrix}, \qquad \begin{pmatrix} W \\ X \\ Y \\ Z \end{pmatrix} = L \begin{pmatrix} H_{i-1} \\ G_{i-1} \\ M_i \\ \overline{Y} \end{pmatrix}$$

*where* $\overline{Y} = E^U_{A\|B}(C)$ *and* $U$ *is a* $4 \times 3$ $\{0,1\}$*-matrix and* $L$ *is a* $4 \times 4$ $\{0,1\}$*-matrix.*

– ***Type-4:*** *For a double-call Rate-1 DBL compression function based on two independent* $(2n, n)$ *blockciphers* $E^U$ *and* $E^L$*, we write the compression function* $(H_i, G_i) = F(H_{i-1}, G_{i-1}, M_i^1, M_i^2)$ *as:*

$$\begin{cases} H_i = f(H_{i-1}, G_{i-1}, M_i^1, M_i^2) = E^U_{A\|B}(C) \oplus D \\ G_i = g(H_{i-1}, G_{i-1}, M_i^1, M_i^2) = E^L_{W\|X}(Y) \oplus Z \end{cases}$$

*where*

$$\begin{pmatrix} A \\ B \\ C \\ D \end{pmatrix} = U \begin{pmatrix} H_{i-1} \\ G_{i-1} \\ M_i^1 \\ M_i^2 \end{pmatrix}, \qquad \begin{pmatrix} W \\ X \\ Y \\ Z \end{pmatrix} = L \begin{pmatrix} H_{i-1} \\ G_{i-1} \\ M_i^1 \\ M_i^2 \\ \overline{Y} \end{pmatrix}$$

*where* $\overline{Y} = E^U_{A\|B}(C)$ *and* $U$ *is a* $4 \times 4$ $\{0,1\}$*-matrix and* $L$ *is a* $4 \times 5$ $\{0,1\}$*-matrix.*

There are some other types such as double-call Rate-1 DBL compression function based on $(n, n)$ block ciphers [13] and its security has been analyzed carefully. In the next section we will show that there doesn't even exist an optimum secure double-call Rate-1 DBL compression function based $(2n, n)$ blockciphers, let alone it is based on $(n, n)$ block ciphers.

## 3 A Criterion for Block-Cipher-Based Compression Functions

In this section we propose a simple criterion for building optimum secure **Type-1, Type-2, Type-3, Type-4** compression functions.

**Theorem 1.** *To build an optimum secure* ***Type-1, Type-2, Type-3, Type-4*** *compression functions, it is required that after elementary column operations on matrix* $L_1$*,* $L_2$*,* $U$*, all the matrix* $L_1$*,* $L_2$*,* $U$ *can be transformed into the following form:*

$$\begin{pmatrix} 0 & 0 & \cdots & 1 \\ 0 & \cdots & 1 & 0 \\ \vdots & & \ddots & \vdots \\ 1 & 0 & \cdots & 0 \\ 1 & ? & \cdots & ? \end{pmatrix}$$

*where* $'?'$ *can be either* $0$ *or* $1$*.*

We will prove this theorem in the next section. Here we explain this theorem by analysis of **Type-1** compression functions. To build an optimum preimage resistant and collision resistant compression function based on a $(n, n)$ blockcipher, Preneel, Govaerts and Vandewalle stated that 12 compression functions are secure against attacks [24]. Here we show there are at most 12 secure compression functions.

**Corollary 1.** *There are at most* 12 *optimum secure* ***Type-1*** *compression functions.*

*Proof.* According to Theorem 1, for each secure **Type-1** compression function, it can be transformed into the following matrix form after elementary column operations on the matrix: $\begin{pmatrix} 0 & 1 \\ 1 & 0 \\ 1 & ? \end{pmatrix}$.

From the matrix operations in Theorem 1, we can compute the matrix for each optimum secure **Type-1** compression function by right-multiplying $L_1$ by a full rank $2 \times 2$ $\{0,1\}$-matrix. According to Lemma 1, there are $2^{2(2-1)/2} \cdot (2^2 - 1) \cdot (2^1 - 1) = 6$ such matrices, since $'?'$ can be either 0 or 1, so there are totally at most $2 \cdot 6 = 12$ optimum secure **Type-1** compression functions. □

**Corollary 2.** *There are at most* 672 *optimum secure* ***Type-2*** *compression functions.*

*Proof.* According to Theorem 1, for each optimum secure **Type-2** compression function, it can be transformed into the following matrix form after elementary column operations on the matrix:

$$\begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & ? & ? \end{pmatrix}$$

For **Type-2** compression functions, $L_2$ is a $4 \times 3$ $\{0,1\}$-matrix, we can compute the matrix for each optimum secure **Type-2** compression function by right-multiplying $L_2$ by a full rank $3 \times 3$ $\{0,1\}$-matrix. According to Lemma 1, there are $2^{3(3-1)/2} \cdot (2^3 - 1) \cdot (2^2 - 1) \cdot (2^1 - 1) = 168$ such matrices, so there are totally at most $2^2 \cdot 168 = 672$ secure **Type-2** compression functions. □

**Corollary 3.** *There are at most* 7676928 *secure* ***Type-3*** *compression functions.*

*Proof.* For **Type-3** compression functions, $U$ is a $4 \times 3$ $\{0,1\}$-matrix and $L$ is a $4 \times 4$ $\{0,1\}$-matrix. The upper blockcipher and the lower block cipher are independent. The matrix $U$ can be transformed into the following form after elementary column operations:

$$\begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & ? & ? \end{pmatrix}.$$

Thus we can right-multiply $U$ by a full rank $3 \times 3$ $\{0,1\}$-matrix. According to Lemma 1, there are 168 full rank $3 \times 3$ $\{0,1\}$ matrices and we can have $2^2 \cdot 168 = 672$ cases of $U$.

It is harder to compute the cases of $L$, since

$$\begin{pmatrix} W \\ X \\ Y \\ Z \end{pmatrix} = L \begin{pmatrix} H_{i-1} \\ G_{i-1} \\ M_i \\ \overline{Y} \end{pmatrix}$$

and thus $F$ can be either parallel or serial [13].

We consider the following two cases of $L$ after elementary matrix operations

$$\begin{pmatrix} W \\ X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} 0\,0\,1\,0 \\ 0\,1\,0\,0 \\ 1\,0\,0\,0 \\ 1\,?\,?\,0 \end{pmatrix} \begin{pmatrix} \overline{Y} \\ G_{i-1} \\ M_i \\ H_{i-1} \end{pmatrix} or \begin{pmatrix} W \\ X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} 0\,0\,1\,? \\ 0\,1\,0\,? \\ 1\,0\,0\,? \\ 1\,?\,?\,? \end{pmatrix} \begin{pmatrix} H_{i-1} \\ G_{i-1} \\ M_i \\ \overline{Y} \end{pmatrix}$$

where in the first case the vector $(X, Y, Z, W)$ is effected by $H_{i-1}$ indirectly through $\overline{Y}$ and in the second case the vector $(X, Y, Z, W)$ is effected by $H_{i-1}$ directly and $\overline{Y}$ is an option. If $L$ cannot be transformed into the above two cases, we can find an attack which is in the proof of the criterion in the next section.

Thus there are $(2^2+2^6) \cdot 168 = 11424$ cases of $L$ and 672 cases of $U$. There are at most $672 \times 11424 = 7676928$ cases of secure **Type-3** compression functions. □

**Corollary 4.** *There are no optimum secure **Type-4** compression functions.*

*Proof.* For **Type-4** compression functions, $U$ is a $4 \times 4$ matrix. A $4 \times 4$ matrix cannot be transformed into the form in Theorem 1, so there doesn't exist any optimum secure **Type-4** compression function. □

## 4 Proof of the Criterion

In this section, we will prove Theorem 1. Without loss of generality, we first consider **Type-2** compression functions, where $H_i = f(H_{i-1}, M_i^1, M_i^2) = E_{A\|B}(C) \oplus D$ and

$$\begin{pmatrix} A \\ B \\ C \\ D \end{pmatrix} = L_2 \begin{pmatrix} H_{i-1} \\ M_i^1 \\ M_i^2 \end{pmatrix}.$$

**Proof of Theorem 1 in Type-2 Compression Functions:**

1. *Fixed Point*: Let $Y = E_{A\|B}(C)$, we call that there exist a fixed point in $f$ if $Y$ is not affected by a linear composition of $(H_{i-1}, M_i^1, M_i^2)$.
   If a fixed point exists in $f$, let $S$ is the linear composition of $(H_{i-1}, M_i^1, M_i^2)$ such that $Y$ is not affected by $S$, then it is trivial to find a (second) pre-image or collision:

- Given $H_i$, we random choose an input $(H_{i-1}, M_i^1, M_i^2)$, compute $Y = E_{A\|B}(C)$, $H_i' = Y \oplus D$, $A$,$B$,$C$ is not affected by $S$ since $Y$ is not affected by $S$.
- Then we fixed $Y$, and change the value of $S$, it is trivial to deduce $(H_{i-1}^{'}, M_i^{1'}, M_i^{2'})$ from $\{Y, H_i, H_i', A, B, C\}$ such that $f(H_{i-1}^{'}, M_i^{1'}, M_i^{2'}) = H_i$. Thus a collision is also trivial to be found.

To build a secure **Type-2** compression functions, it is required such fixed point doesn't exit. That is to say, $A$,$B$,$C$ will be affected by any linear composition of $\{H_{i-1}, M_i^1, M_i^2\}$. In the form of matrix, this means the submatrix of $L_2$ without the last row of $L_2$ has rank = 2. After elementary column operations on the submatrix of $L_2$ without the last row it can be a submatrix in the form Theorem 1 without the last row.

2. *Onewayness:* It doesn't guarantee the onewayness of the compression function if there doesn't exist fixed points. However, onewayness can be easily achieved if we make a little restriction on the last row in $L_2$.
   Assume that after elementary column operations on $L_2$, it is a matrix in the form of Theorem 1. If the first entry in the last row of $L_2$ is 1, it is easy to show the one-way property can be hold. We can easily prove it in the ideal cipher model [1,2].
3. *Collision Resistance and (Second) Preimage Resistance:* A collision consists of two pairs $(H_{i-1}, M_i^1, M_i^2)$ and $(H_{i-1}', M_i^{1'}, M_i^{2'})$ will collide in the same value and the adversary has made the relevant queries to $E$ and/or $D$. The ideal cipher maintains the query lists.
   We show that any forward or inverse query will add at most one item to the query list.
   For a forward query $(A, B, C)$, there is a unique $(H_{i-1}, M_i^1, M_i^2)$ corresponding to this query. $Y = E_{A\|B}(C)$ is close to uniform in $\{0,1\}^n$, thus $H_i = Y \oplus D$ is also close to uniform. Since a blockcipher is a permutation, For the $q$-th fresh query, the output is random chosen from $2^n - (q-1)$ values.
   Similarly, for a $q$-th fresh inverse query $(A, B, Y)$, the ideal cipher outputs a value $C$ chosen from $2^n - (q-1)$ values. From $(A, B, C)$ we obtain a unique $(H_{i-1}, M_i^1, M_i^2)$. Since the first entry in the last row of $L_2$ is 1, thus $H_i$ is linear dependent with $C$ and randomly chosen in $2^n - (q-1)$ values.
   Thus, the probability of a collision after $q$ queries can be bounded by $\frac{1}{2}q(q+1)/(2^n - q)$.
   Similarly, the probability of a (second) preimage can be found after $q$ queries can be bounded by $q/(2^n - q)$. □

We can also easily prove Theorem 1 in the case of **Type-1** compression functions based on the above analysis.

**Proof of Theorem 1 in Type-3 Compression Functions:**

It is a little harder to prove Theorem 1 in **Type-3** compression functions since there are two independent blockciphers.

For **Type-3** compression functions, the output length is $2n$ while the block length of the underlined blockcipher is $n$. $(H_i, G_i) = F(H_{i-1}, G_{i-1}, M_i)$ is:

$$\begin{cases} H_i = f(H_{i-1}, G_{i-1}, M_i) = E^U_{A\|B}(C) \oplus D \\ G_i = g(H_{i-1}, G_{i-1}, M_i) = E^L_{W\|X}(Y) \oplus Z \end{cases}$$

where

$$\begin{pmatrix} A \\ B \\ C \\ D \end{pmatrix} = U \begin{pmatrix} H_{i-1} \\ G_{i-1} \\ M_i \end{pmatrix}, \qquad \begin{pmatrix} W \\ X \\ Y \\ Z \end{pmatrix} = L \begin{pmatrix} H_{i-1} \\ G_{i-1} \\ M_i \\ \overline{Y} \end{pmatrix}$$

where $\overline{Y} = E^U_{A\|B}(C)$ and $U$ is a $4\times 3$ $\{0,1\}$-matrix and $L$ is a $4\times 4$ $\{0,1\}$-matrix.

First we show that $F$ is not optimum collision resistance if $U$ cannot be transformed into the matrix form in Theorem 1. Since $U$ is a $4\times 3$ $\{0,1\}$-matrix, if it is not in the form in Theorem 1 after elementary column operations, then it can be either of the following two cases after elementary column operations:

1. The submatrix of $U$ without the last row of $U$ is not full rank. In this case, there exist fixed points, thus it is trivial to find collisions. After we find $2^{n/2}$ collisions in $H_i$, we input these collisions to the lower part of $F$ and a collision can be found in $G_i$ with probability 0.39.
2. The first entry in the last row is $'0'$. In this case, the function $f$ is not oneway and thus it is trivial to find collisions of $f$. Thus we can easily attack $F$ similarly as the above case.

Thus it is required that the matrix $U$ can be transformed into the matrix form in Theorem 1.

It is harder to analyze the cases of the matrix $L$, since

$$\begin{pmatrix} W \\ X \\ Y \\ Z \end{pmatrix} = L \begin{pmatrix} H_{i-1} \\ G_{i-1} \\ M_i \\ \overline{Y} \end{pmatrix}$$

and thus $F$ can be either parallel or serial [13].

We consider the following two cases of $L$ after elementary matrix operations

$$\begin{pmatrix} W \\ X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} 0\,0\,1\,0 \\ 0\,1\,0\,0 \\ 1\,0\,0\,0 \\ 1\,?\,?\,0 \end{pmatrix} \begin{pmatrix} \overline{Y} \\ G_{i-1} \\ M_i \\ H_{i-1} \end{pmatrix} \textit{ or } \begin{pmatrix} W \\ X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} 0\,0\,1\,? \\ 0\,1\,0\,? \\ 1\,0\,0\,? \\ 1\,?\,?\,? \end{pmatrix} \begin{pmatrix} H_{i-1} \\ G_{i-1} \\ M_i \\ \overline{Y} \end{pmatrix}$$

where in the first case the vector $(X, Y, Z, W)$ is effected by $H_{i-1}$ indirectly through $\overline{Y}$ and in the second case the vector $(X, Y, Z, W)$ is effected by $H_{i-1}$ directly and $\overline{Y}$ is an option.

If $L$ can not be transformed into the above two cases after elementary column operations, there exist fixed points such that if we fix the linear composition of $(H_{i-1}, G_{i-1}, M_i)$, then $H_i$ is fixed, thus it is trivial to find collisions.

*Collision Resistance and (Second) Preimage Resistance:*

$(H_i, G_i) = F(H_{i-1}, G_{i-1}, M_i) = f(H_{i-1}, G_{i-1}, M_i) \parallel g(H_{i-1}, G_{i-1}, M_i)$.

The upper cipher $E^U$ in $f$ and lower cipher $E^L$ in $g$ are independent. If $L$ can be transformed into the first case, we can rewrite $F$ as $F(x_1, x_2, x_3) = f(x_1, x_2, x_3) \parallel g(f(x_1, x_2, x_3) \oplus x_1, x_2, x_3)$ where $(x_1, x_2, x_3) = (H_{i-1}, G_{i-1}, M_i)$ and $f$ , $g$ are two independent compression functions.

If $f$ and $g$ are Davies-Meyer structure, we can prove the collision resistance and (second)preimage resistance bound in the ideal cipher model as in [10].

It is even easier to analyze the collision resistance and (second) preimage resistance in the ideal cipher model when $L$ can be transformed into the second case, thus we omit it here.

Thus for compression function $F$, the probability of a collision after $q$ queries can be bounded by $q^2/(2^n - q)^2$. and the probability of a (second) preimage can be found after $q$ queries can be bounded by $q/(2^n - q)^2$.

□

For **Type-4** compression functions, the matrix $U$ is a $4 \times 4$ matrix. A $4 \times 4$ matrix cannot be transformed into the form in Theorem 1, so there doesn't exist any optimum secure **Type-4** compression function.

Although the compression functions are not optimum secure against collision attack and (second)preimage attack, one can build a collision resistant hash function by iterating the compression function. In [28], Steinberger proves a strong security bound for MDC-2. Although there exists a collision attack on the compression function needs $2^{n/2}$ queries, after the iteration, the collision resistance can be improved to $2^{3n/5}$ queries. The compression function of MDC-2 is not any type we discussed above, since it splits the upper value and lower value and diffuse them.

In the next section, we will show that Type-4 hash functions are not optimum collision resistance even in the iteration, thus we solve Hirose's open problem that there may exist some Type-4 hash functions are optimum collision resistance through iteration [9].

## 5 Attacks on Type-4 Hash functions in the Iteration

In this section we find attacks on Type-4 Hash functions in the Iteration. The initial value is randomly chosen and fixed. We answer the open problem proposed by Hirose [9] that there does not exist optimum collision resistant **Type-4** hash functions through iteration.

**Theorem 2.** *If we iterated **Type-4** compression function in the Merkle-Damgard model and fixed the initial value, there exist 2nd preimage and preimage attacks with complexity of about $4 \times 2^n$. Furthermore, there exists a collision attack with complexity of about $3 \times 2^{3n/4}$.*

*Proof.* For **Type-4** compression functions,

$$\begin{pmatrix} A \\ B \\ C \\ D \end{pmatrix} = U \begin{pmatrix} H_{i-1} \\ G_{i-1} \\ M_i^1 \\ M_i^2 \end{pmatrix}$$

$U$ is a $4 \times 4$ matrix and cannot be converted into the form in Theorem 1. There must exist a linear composition of $\{H_{i-1}, G_{i-1}, M_i^1, M_i^2\}$ that don't effect the value of $A, B, C$. If the initial value $H_0, G_0$ is fixed in the Merkle-Damgård iteration, we have the following meet-in-the-middle attacks, which is from Knudsen et al.'s idea [13]:

- The attacks searches for the four message blocks $(M_1^1, M_1^2)$ and $(M_2^1, M_2^2)$ such that the hash result is hit in $(H_2, G_2)$ (in the case of a (2nd) preimage attack) or for a pair of four correcting blocks which yield a collision.
- The (2nd) preimage attack:
  1. Backward step: choose $2^n$ values of $(A, B, \overline{Y})$, query to the blockcipher $E^U$ and obtain $C$ and compute $H_1', G_1', M_2^{1'}, M_2^{2'}$ from $A, B, C, \overline{Y}, H_2$, this requires at most $2^n$ queries to the upper blockcipher $E^U$.
     It should be noted that there exists a special case that the adversary needs to query $2^n$ times to find a correct $(A, B, \overline{Y})$ corresponding to $H_2$, but after that, the attacker can find $2^n$ preimages to $H_2$ without any additional queries. For example, if $H_2 = E^U_{M_1^1 \| M_1^2}(H_1 \oplus G_1) \oplus H_1 \oplus G_1$, $(A, B, C, \overline{Y}) = (M_1^1, M_1^2, H_1 \oplus G_1, H_1 \oplus G_1 \oplus H_2)$, given $H_2$, the attacker needs to query $2^n$ times to find a correct $(A, B, C, \overline{Y})$ to satisfy the condition that $\overline{Y} = C \oplus H_2$.
     But if we obtain such a pair $(A, B, C)$ satisfies the condition, it is trivial to find $2^n$ preimages to $H_2$ without any additional queries to $E^U$, since there are $2^n$ choices of $H_1$ and $G_1$ such that $C = H_1 \oplus G_1$.
  2. Forward step: choose $2^n$ values for $(M_1^1, M_1^2)$ and compute $(H_1, G_1)$ from $(H_0, G_0)$.
  3. Find matches $H_1^1 = H_1'$. For every match we compute the corresponding value of $G_2$. The quantities in the meet-in-middle attack are $n$ bits long, so it gives about $\frac{2^n \times 2^n}{2^n} = 2^n$ values of $(H_1, G_1, M_2^1, M_2^2)$ all hitting the same value of $H_2$. Thus, $G_2$ will be found with probability about 0.63; the total number of operations is about $4 \times 2^n$.
- The collision attack:
  1. Backward step: choose $2^{3n/4}$ values of $(A, B, C)$ and compute $H_1'$, $G_1'$, $M_2^{1'}$, $M_2^{2'}$ from $A, B, C, H_2$.
     It should be noted that there exists special cases just discussed in (2nd) preimage attack. For such cases, it is trivial to find many collisions which lead to a same value without any queries to the blockcipher $E^U$. Take the same example above, the attacker can fixed $H_1 \oplus G_1$ but change $H_1$ or $G_1$, the hash value $H_2$ will never change. In this case, one just choose $2^{3n/4}$ values of $H_1'$ and $G_1'$ with the same $(M_2^{1'}, M_2^{2'})$.

2. Forward step: choose $2^{3n/4}$ values for $(M_1^1, M_1^2)$ and compute $(H_1, G_1)$ from $(H_0, G_0)$.
3. Find matches $H_1^1 = H_1'$. For every match we compute the corresponding value of $G_2$. The quantities in the meet-in-middle attack are $n$ bits long, so it gives about $\frac{2^{3n/4} \times 2^{3n/4}}{2^n} = 2^{n/2}$ values of $(H_1, G_1, M_2^1, M_2^2)$ all hitting the same value of $H_2$. Thus, a collision of $G_2$ will be found with probability about 0.39; the total number of operations is about $3 \times 2^{3n/4}$.

□

In [9], Hirose left an open problem that if the following two compression functions are optimally collision resistant in the iteration:
**Case 1:**

$$H_i = E^U_{M_i^1 \| M_i^2}(H_{i-1} \oplus G_{i-1}) \oplus H_{i-1} \oplus G_{i-1}.$$
$$G_i = E^L_{M_i^1 \| M_i^2}(H_{i-1}) \oplus H_{i-1}.$$

**Case 2:**

$$H_i = E^U_{M_i^1 \| M_i^2}(H_{i-1}) \oplus G_{i-1}.$$
$$G_i = E^L_{M_i^1 \| M_i^2}(G_{i-1}) \oplus H_{i-1}.$$

Based on the proof in Theorem 2, there exist a $3 \times 2^{3n/4}$ collision attack on these two DBL hash functions in the iteration.

**Improve the collision resistance in the Iteration:**
If we modify the compression function slightly, just as the MDC-2 design, that is, we write $H_i = H_i^1 \parallel H_i^2$, $G_i = G_i^1 \parallel G_i^2$, where $H_i^1, H_i^2, G_i^1, G_i^2$ are $n/2$ bits, the final output of the compression function is $(H_i^1 \parallel G_i^1, H_i^2 \parallel G_i^2)$, the attack in Theorem 2 failed. Thus we think such slightly modifications can improve the collision resistance of the hash function in the iteration.

**DBL Constructions Based on one Blockcipher:** For DBL compression functions discussed above, only **Type-3** constructions can achieve the optimum security and two independent blockciphers are needed. However, it is practical to use only one blockcipher. There exists many ways to construct another (pseudo) independent blockcipher from a blockcipher $E_K(X)$, such as $E_{K \oplus c}(X)$, $E_K(X \oplus c)$ where $c$ is a nonzero constant. This method has been adopted in the design of Abreast-DM [15] and Hirose's Scheme [10]. In the design of Tandem-DM [15], we can assume the upper blockcipher and lower blockcipher are (pseudo) independent except a negligible probability since the input to the key of the lower blockcipher cannot be chosen by the adversary but determined by the output of the upper blockcipher.

## 6 A Problem in Stam's Theorem

In [27], Stam proposed a Rate-1 compression function that is optimally collision resistant up to a logarithmic factor. Stam's construction only based on a single

block cipher and only nees one cipher call. He named this construction as the supercharging construction. He proposed the following definition:

**Supercharged compression function.** *A single call blockcipher based compression function $H^E$ is called supercharged single call Type-I with overlap $\gamma$ iff $s \geq n$, $m + s = n + k$ and the following three hold:*

1. *The preprocessing $C^{PRE}$ is bijective.*
2. *For all $M, V$ the postprocessing $C^{POST}(M, V, \cdot)$ is injective, with effective range $R_{POST,(M,V)}$.*
3. *For all $K, Y$ the modified postprocessing $C^{AUX}(K, \cdot, Y)$ is injective, with effective range $R_{AUX,(K,Y)}$.*

*Where the overlap $\gamma$ is defined as: $\gamma = max|R_Z \cap R_{Z'}| : Z, Z' \in \{POST, AUX\} \times 0, 1^{k+n}, Z \neq Z'$.*

Based on this definition, Stam proposed a theorem about the collision resistance of the supercharged compression function and the following corollary:

**Stam's Corollary:** *Let $H^E$ be a supercharged single call Type-I compression function with overlap $\gamma$. Then for $q < 2^{n-1}/\gamma^{\frac{1}{2}}$ the probability of finding a collision can be upper bounded by $Adv_H^{coll}(q) \leq 2max\ (2e\gamma^{\frac{1}{2}}, m+n+s+2)q/2n$.*

Here we proposed a counterexample which is a superchareged single call blockcipher based compression with overlap 3 and $m = n, s = k = 2n$. Based on Stam's Theorem, the collision resistance of such compression functions is: for $q \leq 2^{n-\frac{3}{2}}$,

$$Adv_H^{coll}(q) \leq (n + \frac{1}{2})q/2^{n-3}.$$

**A Counterexample:** Let $(+, \cdot)$ be the addition and multiplication over $\mathbb{F}_{2^n}$. For a $(2n, n)$ ideal blockcipher $E$, we construct a DBL compression function $(W_1, W_2) = F(V_1, V_2, M)$ as follows:

1. Set $K \leftarrow (V_1, V_2)$ and $X \leftarrow M$.
2. Compute $Y \leftarrow E_K(X)$.
3. Compute $W_1 \leftarrow Y + M$ and $W_2 \leftarrow M \cdot W_1^2 + V_1 \cdot W_1$; output $(W_1, W_2)$.

Now we see that this construction is a supercharged single call Type-I compression function with overlap $\gamma = 3$:

1. $C^{PRE}(M, V_1, V_2) = (V_1 \parallel V_2, M)$ is bijective.
2. For all $M, V_1, V_2$ the postprocessing $C^{POST}(M, V_1, V_2, y) = (y + M, M \cdot (y + M)^2 + V_1 \cdot (y + M))$ is injective in $y$.
3. For all $K, Y$ the modified postprocessing $C^{AUX}(K, x, Y) = (Y + x, x \cdot (Y + x)^2 + V_1 \cdot (Y + x))$ is injective in $x$.
4. It is easy to see

$$R_{POST,(M,V_1,V_2)} = \{(W, M \cdot W^2 + V_1 \cdot W) | W \in \{0, 1\}^n\}$$

and with a little bit more effort, using that $M = Y + W$ and $K = (K_1, K_2) = (V_1, V_2)$,

$$R_{AUX,(K_1,K_2,Y)} = \{(W, W^3 + Y \cdot W^2 + K_1 \cdot W) | W \in \{0,1\}^n\}$$

As a result, for $(W_1, W_2)$ to be the intersection of $R_{POST}$ and $R_{AUX}$, we require $W_1$ to be a root of the difference of the two polynomials that define $W_2$ for $R_{POST}$ and $R_{AUX}$. It can be readily verified that the relevant two polynomials are distnct, and the resulting difference is a non-zero polynomial of degree at most three. It will therefore have at most three roots over $\mathbb{F}_{2^n}$, thus $\gamma = 3$.

Based on Stam's corollary, for $q \leq 2^{n-\frac{3}{2}}$, the upper bound for collision resistant is

$$Adv_H^{coll}(q) \leq (n + \frac{1}{2})q/2^{n-3}.$$

However, it is easy to see that there exists a collision attack only needs $2^{n/2}$ queries. The attacker fixes $M, V_1$ and randomly choose $2^{n/2}$ values of $V_2$, with probability 0.39 he obtains a collision $(M, V_1, V_2)$ and $(M, V_1, V_2')$ collide on $W_1$, since $W_2 = M \cdot W_1^2 + V_1 \cdot W_1$ which is not effected by $V_2$ and $V_2'$, he also obtains a collision at $W_2$. The attacker only needs $2^{n/2}$ queries, which is much less than Stam's corollary.

Through this counterexample, one can see that there exists flaws in Stam's definition of **Supercharged compression function**. The three conditions in the definition are not enough to guarantee the collision bound claimed by Stam. Here we add another condition:

**The 4th condition:** *If the the supercharged compression function can be write as:*

$$W_1 = F_1(V_1, V_2, M)$$
$$W_2 = F_2(V_1, V_2, M, W_1)$$

*there should not exist fixed points, that is, $W_1$ is effected by any linear composition of $(V_1, V_2, M)$ and $W_2$ is also effected by any linear composition of $(V_1, V_2, M)$ even if $W_1$ is fixed.*

Combined this condition with Stam's conditions, one can easily use Stam's proof to prove the collision bound for supercharged compression functions, which is the same as Stam's work.

Through on our analysis, one should be very careful when he claimed to give a security proof to a construction. Although the theorem is right sometimes, the proof has flaws, just as stated in [18].

## 7 Conclusion

In this paper We gave a new definition (security rate) for blockcipher-based hash functions. This new definition is different from the efficiency rate which

only measures the efficiency of a blockcipher-based hash functions. We proposed a criterion for the optimum security of blockcipher-based hash functions and this criterion is simpler than previous work in the literature. Based on this criterion, we counted the exact number of optimum security **Type-1, Type-2, Type-3, Type-4** compression functions.

There are 12 **Type-1**, 672 **Type-2**, 7676928 **Type-3** optimum secure compression functions and we proved their security in the ideal cipher model. We pointed out that Abreast-DM, Tandem-DM and Hirose's Scheme are just special cases of **Type-3** compression functions. Thus they are optimum secure in the ideal cipher model.

Based on the criterion, there doesn't exist any **Type-4** optimum secure compression functions. We also found collision attacks on **Type-4** hash functions in the iteration with complexity much lower than the birthday complexity. Thus we solved the open problem left by Hirose that if there exists optimum collision resistant **Type-4** hash functions in the iteration.

At last, we pointed out a flaw in Stam's theorem and added another condition for the security of supercharged compression functions. Through our analysis, it seems optimum collision resistance and (2nd) preimage resistance Rate-1 DBL compression functions doesn't exist.

## References


1. J. Black, P. Rogaway, and T. Shrimpton. Black-box analysis of the block-cipher-based hash-function constructions from PGV. In *Advances in Cryptology - Crypto 2002, Proceedings*, volume 2442 of *Lecture Notes in Computer Science*, pages 320–335. Springer-Verlag, 2002.
2. J. Black, P. Rogaway, T. Shrimpton, and M. Stam. An analysis of the blockcipher-based hash functions from PGV. *Journal of Cryptology*, 23(4):519–545, 2010.
3. FIPS. FIPS 180-1 Secure Hash Standard. Federal Information Processing Standard (FIPS), Publication 180-1, National Institute of Standards and Technology, US Department of Commerce, Washington D.C, 1995.
4. E. Fleischmann, M. Gorski, and S. Lucks. Security of cyclic double block length hash functions. In *Cryptography and Coding 2009*, volume LNCS 5921, pages 153–175. Springer-Verlag, 2009.
5. Ewan Fleischmann, Michael Gorski, and Stefan Lucks. On the security of tandem-DM. volume 5665 LNCS of *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, pages 84–103, Leuven, Belgium, 2009. Springer Verlag.
6. M. Hattori, S. Hirose, and S. Yoshida. Analysis of double block length hash functions. *Cryptography and Coding, Proceedings*, 2898:290–302, 2003.
7. S. Hirose. Provably secure double-block-length hash functions in a black-box model. *Information Security and Cryptology - Icisc 2004*, LNCS 3506:330–342, 2004.
8. S. Hirose. Provably secure double-block-length hash functions in a black-box model. volume 3506 of *Lecture Notes in Computer Science*, pages 330–342, Seoul, Korea, Republic of, 2005. Springer Verlag.
9. S. Hirose. A security analysis of double-block-length hash functions with the rate 1. *Ieice Transactions on Fundamentals of Electronics Communications and Computer Sciences*, E89A(10):2575–2582, 2006.

10. S. Hirose. Some plausible constructions of double-block-length hash functions. In *Fast Software Encryption*, volume LNCS 4047, pages 210–225, 2006.
11. Walter Hohl, Xuejia Lai, Thomas Meier, and Christian Waldvogel. Security of iterated hash functions based on block ciphers. In *Advances in Cryptology - CRYPTO'93*, volume LNCS 773, pages 379–379, Santa Barbara, CA, United states, 1994. Springer-Verlag.
12. ISO. ISO/IEC 10118 Information technology - Security techniques - Hash-functions, 1994.
13. L. R. Knudsen, X. J. Lai, and B. Preneel. Attacks on fast double block length hash functions. *Journal of Cryptology*, 11(1):59–72, 1998.
14. X. Lai. *On the design and security of block ciphers*, volume 1 of *ETH Series in Information Processing*. Hartung-Gorre Verlag, Konstanz, 1992.
15. X. Lai and J. L. Massey. Hash functions based on block ciphers. In R. A. Rueppel, editor, *Advances in Cryptography-Eurocrypt'92*, volume LNCS 658, pages 55–70. Springer-Verlag, 1992.
16. R. Lidl and H.Niederreiter. *Finite fields*. Encyclopedia of Mathematics and its applications. Addision-Wesley Publishing Company, 1983.
17. S. Lucks. A collision-resistant rate-1 double-block-length hash function. In *Symmetric Cryptography, number 07021 in Dagstuhl Seminar Proceedings*, Dagstuhl, Germany, 2007. Internationales Begegnungs - und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Germany.
18. Yiyuan Luo, Zheng Gong, Ming Duan, Bo Zhu, and Xuejia Lai. Revisiting the indifferentiability of PGV hash functions. Cryptology ePrint Archive, Report 2009/265, 2009. `http://eprint.iacr.org/`.
19. R. C. Merkle. One way hash functions and DES. In *Advances in Cryptology - CRYPTO'89*, volume LNCS 435, pages 428–446. Springer-Verlag, 1989.
20. M. Nandi. Towards optimal double-length hash functions. In *INDOCRYPT'05*, volume LNCS 3797, pages 77–89. Springer-Verlag, 2005.
21. M. Nandi, W. Lee, K. Sakurai, and S. Lee. Security analysis of a 2/3-rate double length compression function in the black-box model. In *Fast Software Encryption - FSE'2005*, volume LNCS 3557, pages 243–254. Springer-Verlag, 2005.
22. NIST. Third (final) round candidates, 2010. http://csrc.nist.gov/groups/ST/hash/sha-3/Round3/submissions-rnd3.html.
23. O. Özen and M. Stam. Another glance at double-length hashing. In *Cryptography and Coding, 12th IMA International Conference, Cryptography and Coding 2009*, volume LNCS 5921, pages 176–201. Springer-Verlag, Berlin, 2009.
24. B. Preneel, R. Govaerts, and J. Vandewalle. Hash functions based on block ciphers: A synthetic approach. In D.R. Stinson, editor, *Advances in Cryptology - Proc. Crypto'93*, volume LNCS 773, pages 368–378. Springer-Verlag, Berlin, 1993.
25. R. L. Rivest. The MD5 message digest algorithm. In *Request for Comments (RFC) 1321*. Internet Activities Board, Internet Privacy Task Force, 1992.
26. Takashi Satoh, Mio Haga, and Kaoru Kurosawa. Towards secure and fast hash functions. *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences*, E82-A(1):55–62, 1999.
27. M. Stam. Block cipher based hashing revisited. In *Fast Software Encryption 2009*, volume LNCS 5665, pages 67–83. Springer, Berlin, 2009.
28. John P. Steinberger. The collision intractability of MDC-2 in the ideal-cipher model. In *Advances in Cryptology-Proceedings of EUROCRYPT 2007*, volume LNCS 4515 of *Lecture Notes in Computer Science*, pages 34–51, Barcelona, Spain, 2007. Springer Verlag, Berlin.

29. Xiaoyun Wang, Yiqun Lisa Yin, and Hongbo Yu. Finding collisions in the full SHA-1. In Victor Shoup, editor, *Advances in Cryptology - CRPTO'05*, volume LNCS 3621, pages 17–36, Santa Barbara, CA, USA, 2005. Springer-Verlag.
30. Xiaoyun Wang and Hongbo Yu. How to break MD5 and other hash functions. In Ronald Cramer, editor, *Advances in Cryptology - EUROCRYPT'05*, volume LNCS 3494, pages 19–35, Aarhus, Denmark, 2005. Springer-Verlag.