Turkish Journal of Science & Technology
17(2), 197-201, 2022

Research Paper
https://doi.org/10.55525/tjst.1140811

# Mathematical Analysis of The Hash Functions as a Cryptographic Tools for Blockchain

**Muharrem Tuncay GENÇOĞLU**[1*]
[1] Firat University, Vocational School of Technical Sciences, Elazı ˘g, Turkey
[*1] mtgencoglu23@gmail.com

**(Geliş/Received: 05/07/2022; Kabul/Accepted: 02/09/2022)**

**Abstract-** Blockchain is one of the most interestingly developing technologies today, with its applications in many fields from smart contracts to cryptocurrencies. In this respect, blockchain is a hot modern topic nowadays. This study presents a mathematical analysis of cryptographic hash functions, which are one of the most important elements for understanding the security foundations of this technology. In this analysis presented; Hash functions, which are one of the building blocks of blockchain technology, used to ensure information integrity, have proven to be resistant to collision resistance, which is very important in data mining. Thus, a theory has been put forward that will contribute to time and energy saving, which is one of the important problems in data mining, in which blockchain technology is used.

**Keywords:** hash functions, blockchain, cryptographic hash functions, mathematical analysis

## Blok Zincir için bir Kriptografik Araç Olarak Hash Fonksiyonlarının Matematiksel Analizi

**Öz:** Blockchain, akıllı sözleşmelerden kripto paralara kadar birçok alanda uygulamalarıyla günümüzde en ilginç gelişen teknolojilerden biridir. Bu bağlamda, blockchain günümüzde sıcak ve modern bir konudur. Bu çalışma, bu teknolojinin güvenlik temellerini anlamak için en önemli unsurlardan biri olan kriptografik özet fonksiyonlarının matematiksel bir analizini sunmaktadır. Sunulan bu analizde; Bilgi bütünlüğünü sağlamak için kullanılan blockchain teknolojisinin yapı taşlarından biri olan hash fonksiyonlarının veri madenciliğinde oldukça önemli olan çarpışma direncine karşı dirençli olduğu kanıtlanmıştır. Böylece blockchain teknolojisinin kullanıldığı veri madenciliğinde önemli sorunlardan biri olan zaman ve enerji tasarrufuna katkı sağlayacak bir teori ortaya konulmuştur.

**Anahtar kelimeler:** Hash fonksiyonları, blok zincir, kriptografik hash fonksiyonları, matematiksel analiz

## 1. Introduction

Blockchain is one of the most popular and developing technologies in recent years. This technology is briefly; It can be defined as an interconnected chain of data blocks that allows the creation of transaction records based on a managed distributed consensus protocol without a central authority. Thanks to this structure, any of the participants cannot change the content of the blocks that have been agreed upon. Therefore, only new transactions can be added or merged to eliminate or modify existing ones. In this structure, three basic features complete the principle of immutability;

1. A summary of the state of the entire chain should be available at any time to prevent and detect manipulation of any block of the chain.
2. It is necessary to verify whether a transaction is included in the blockchain.
3. Parties involved in a transaction in any block should be allowed to do so in a so-called anonymous manner.

Conceivably one of the best-known applications of this technology, bitcoin, the cryptocurrency so named for its use makes several cryptographic primitives, and ensures the pseudo-anonymity of participants, and the immutability of stored records, and distributed consensus without recourse to a central authority[1].

In this study, the most basic cryptographic concept of blockchain technology and the concept of the hash function, which is the primary tool in providing information integrity, are reviewed. In addition, the mathematical proof that it is a collision-resistant function is presented.

The remainder of this article is organized as follows; In the second chapter, the definition and properties of the hash function used in verifying data integrity and the definition and properties of the concept of modular arithmetic are given. In the third part, a theorem that can be a solution to the computational problem of data miners

* Sorumlu yazar: mtgencoglu23@gmail.com. Yazarların ORCID Numarası: [1]0000 -0002-8784-9634

using blockchain technology, which includes finding partial collisions of a certain hash function, has been proved. Finally, the results of the study are presented in the fourth chapter.

## 2. Hash Functions

Hash functions are among the cryptographic primitives that have grown in relevance in recent years. It is important to point out that such functions do not encrypt or decrypt messages. However, they are an indispensable tool for verifying data integrity, apart from other applications equally interesting, hash functions can be defined as functions that are capable of transforming any block of binary data into another fixed-size binary block**[**2,3**].** The result of such a transformation is called has hor digest.

In addition to this initial use, hash functions have been applied to other areas concerned with the protection of information in general, and its integrity in particular. Therefore, it can detect corrupt data, the presence of viruses, etc. They are also used to detect.

One-way functions that take any variable-length input and convert it to a fixed-length output are called hash functions[3].

The hash function is the operation that creates a fixed-length unique value with mathematical functions of various lengths of data. In other words, hash functions, which have a very important place in cryptography, compress data of any size to a fixed length. It is a one-way function and although a relationship is established between the processed text and the summary value, the original data cannot be obtained from the summary value. In the hashing process, the same value is produced for the same data, but when there is the slightest chance, the value created by the hash function changes. Hash functions are widely used in areas such as verifying the integrity of data, storing passwords, digital signatures, message authentication code, and blockchain.

In hash functions, different inputs can produce the same output. This situation is called conflict. This is not desirable for hash functions and compromises their security. Hash functions have advantages such as guaranteeing data integrity, producing fixed and small-sized outputs, and producing fast output for each input length.

Mathematically, the hash function is expressed using the concept of a One Way Function.

m; variable size and predetermined message,

M; a particular set that gives a summary of the message m

n; dimension provided that;

$$\acute{h}: M \rightarrow \{0,1\}^n,$$

$$h(m) = \acute{m}.$$

In other words; At $Y = f(x)$, They are functions that are easy to find when x is known, but difficult to find when Y is known. Hash functions with this property are collision-proof functions.

### 2.1. Properties

For an ideal cryptographic hash function to be considered secure, it should have the following three properties[4];

I. *Collision Resistance:* Messages should be difficult to find when the digest of two separate messages is the same.

For $m_1 \neq m_2$, when $\acute{m}_1 = \acute{m}_2$, calculating $m_1, m_2$ should be difficult.

II. *Inverse Image resistance:* The original data cannot be found from the hash function generated by the hash function. That is, it should be difficult to find m when the summary $\acute{m}$ is given.

$f^{-1}(\acute{m})$ must not be computable.

III. *Secondary Reverse Image Resistance:* It must be very difficult to have the same digest of two separate messages.

For $m_1 \neq m_2$, it should be $\acute{m}_1 \neq \acute{m}_2$.

As stated earlier, blockchains are chains of information blocks where each block has an associated hash value. These hashes are used to create a data index. If each data block contains the hash of the block previously

added to the chain, a linked blockchain is obtained, where the hashes play the role of pointers. In this context, it is important to define a method by which it is possible to determine whether a particular block has been obtained. This can be done with the hash function[5].

For quite some time, the most widely-used hash function was MD5 (Message Digest 5), proposed by Rivest[4], which generates hashes of 128 bits. However, when some security vulnerabilities were published in 2005, the use of the MD5 function was discontinued[6]. Although still used in insecure contexts such as checking the integrity of downloaded files, it is prohibited in environments where security is a critical element. The SHA-1 (Secure Hash Algorithm-1) function, which produces 160-bit hashes, was adopted by the National Institute of Standards and Technology (NIST) in 1995[7 Although there are collisions, SHA-1 continues to be used quite frequently today[8]. As in the MD5 example, its use is discarded by most international institutions and organizations within the scope of secure applications. Another function that provides 160-bit hashes and is still used in some scenarios today is the RIPEMD-160 (RACE Integrity Primitives Assessment Message Digest) function[9]. The SHA-2 family of hash functions is the successor to the SHA-1 function. The SHA-2 specification includes the functions SHA-224, SHA-256, SHA-384, and SHA-512, which provide hashes of 224, 256, 384, and 512 bits, respectively; where SHA-224 and SHA-384 are shortened versions of SHA-256 and SHA-512 functions, respectively. This range of hash sizes significantly improves security compared to the output lengths of MD5 and SHA-1[6].

### 2.2. Modular Arithmetic

The set of remaining classes according to the module m, $m \in z^+$,

$^z/_m = \{\bar{0}, \bar{1}, \bar{2}, \dots, \overline{m-1}\}$,

$^z/_n = \{\bar{0}, \bar{1}, \bar{2}, \dots, \overline{n-1}\}$.

*Explanation:*

If $x, y \in z$ , $x \equiv y \ (mod\ m)$,

1. The remainder of x and y divided by m are equal.

2. The difference between x and y is divisible by m.$x \equiv y \ (mod\ m) \ \Rightarrow x - y = k.m$,

3. The remainder in the division of a sum equals the sum of the individual remainders.

$\overline{x+y} = \bar{x} + \bar{y}$.

4. The remainder in the division of multiplication is equaled to the multiplication of the individual remainders. $\overline{x.y} = \bar{x}.\bar{y}$.

5. $\left.\begin{matrix} a \equiv b \ (mod\ m) \\ c \equiv d \ (mod\ m) \end{matrix}\right\} \Rightarrow \begin{matrix} a + c \equiv b + d \ (mod\ m) \\ a.c \equiv b.d \ (mod\ m) \end{matrix}$ ,

6. $a \equiv b \ (mod\ m)$ and $k \in z \ \Rightarrow a \equiv b \pm km$.

7. $a \equiv b \ (mod\ m)$ and $n \in z^+$ while

$a^n \equiv b^n \ (mod\ m)$.

### 3. Collisions Problem of Hash Functions

Miners using blockchain technology are accountable for figüre out the computationally arduous problem, which involves finding partial collisions of a given hash function. A collision consists of finding two different messages $m_1 \, and \, m_2, m_1 \neq m_2$, such that $\widetilde{m}_1 = \widetilde{m}_2$. In this case, the chosen hash function is SHA-256[6].

To find a partial collision, each miner takes into account the values (titles and transactions) that make up a block, then uses the difficulty corresponding to the moment he did his computations and tries different nonces until the resulting hash value is lesser a certain threshold. This threshold value is equivalent to a certain number of leading zero hashes. Such several zeros are determined by the number of miners in the network (and their computing power), so the time required to find a solution is about 10 minutes on average. It is obvious that the higher the number of leading zeros, the more difficult it is to solve the problem.

The mathematical problem of finding partial collisions guarantees not to use of the same block. This is known in the cryptocurrency literature as Proof of Work (PoW)[10,11].

### 3.1. Theorem

While p is a prime number, $q = \frac{p-1}{2}$ is also prime. This situation is called absolute primality.
α and β, primitive roots of $\alpha^a \equiv \beta \ (mod\ p)$,
$a$; being unknown
Let a hash function be defined as $h$: ${}^{Z}/_{q^2 Z} \rightarrow {}^{Z}/_{pZ}$. This function, for the message
$m = r_0 + r_1 q \qquad (0 \leq r_0 , r_1 \leq q - 1)$ expressed as $h(m) = \alpha^{r_0}\beta^{r_1} \ (mod\ p)$. This h function is highly resistant to collision resistance.

### 3.2. Proof

Let's assume that; for $m = r_0 + r_1 q \quad ve\ m_1 = {r_0}' + {r_1}' q$,

$\alpha^{r_0}\beta^{r_1} \equiv 1 \ (mod\ p)$. (1)

Since $\beta \equiv \alpha^a (mod\ p)$ the above equivalence can be written as

$\alpha^{a(r_1 - {r_1}') - ({r_0}' - r_0)} \equiv 1 \ (mod\ p)$. (2)

However, since there is a primitive root in $\alpha \ mod\ p$, the following result is obtained;

$\alpha^b \equiv 1 \ (mod\ p) \Leftrightarrow b = 0 \ (mod\ p - 1)$. Therefore;

$a(r_1 - {r_1}') \equiv ({r_0}' - r_0) \ (mod\ p - 1)$.

If $g = OBEB(r_1 - {r_1}', p - 1)$, then using the properties of modular arithmetic we can say that equivalence (2) has exactly $g$ solutions[2].

Since $\frac{p-1}{2}$ is prime and $0 \leq r_1 , {r_1}' \leq q - 1$, it must be $-(q-1) \leq r_1 - {r_1}' \leq q - 1$.

So if $r_1 - {r_1}' \neq 0$ then $q > |r_1 - {r_1}'|$. Therefore, there are solutions $g_1\ and\ g_2$.

From the equivalence $\alpha^a \equiv \beta$, we can say that only two possible values of $a$, and one of them will give $\beta$. If we can show that is hidden, that is, unknown, of $a$, we have proved the theorem.

$r_1 - {r_1}' \Rightarrow r_0 = {r_0}' \ (mod\ p - 1)$ becomes$r_0 \equiv {r_0}'$ from the inequality $-(q-1) \leq r_1 - {r_1}' \leq q - 1$ means. This indicates that massages $m\ ve\ m_1$ are the same. That is, $m = m_1$, which contradicts the assumption that messages $m\ ve\ m_1$are different. Then it is $r_1 - {r_1}' \neq 0$. This indicates that $a$ is hidden, that is, unknown. Therefore, the function $h(m) = \alpha^{r_0}\beta^{r_1} \ (mod\ p)$ is a strong collision-resistant function.

## 4. Conclusions

As a result; It is thought that this study will contribute to the solution of the computationally difficult problem, which includes finding the collisions of hash functions, which is very important for data mining in which the blockchain technology used in the crypto money system, which is very popular today, is very important. Also, miners compare the obtained hashes using one-time keys to find partial collisions of a particular hash function. They spend a lot of time and energy making these comparisons. Considering the methods and concepts used in mathematical analysis, they will save time and therefore save energy.

When dealing with blockchains, hash functions appear in a few places. On the one hand, they are used as part of numerical signature algorithms. On the other hand, hash functions play a significant role in implementing Merkle trees, a notion that allows to efficiently check whether a particular block exists in a blockchain. Thanks to the combination of these elements, it has been possible to form a promising trustworthy technology - a blockchain.

Finally; We have shown mathematically that the above-mentioned partial collisions can be found mathematically, namely PoW, and that double spending can be detected in a short time.

## References

[1] Martínez, V.G., Hernández-Álvarez, L., and Hernández Encinas, L, Analysis of the Cryptographic Tools for Blockchain and Bitcoin, Mathematics 2020; 8: 131.
[2] Menezes, A.J.; van Oorschot, P.C.; Vanstone, S.A. Handbook of Applied Cryptography; USA: CRC Press, Inc.: Boca Raton, FL, 1996.
[3] Paar, C., Pelzl, J. Understanding Cryptography. A Textbook for Students and practitioners; Germany: Springer-Heidelberg, 2010.
[4] Rosen, K. An INTRODUCTION to CRYPTOGRAPHY. USA: Taylor&Francis Group, 2007.
[5] Merkle, R.C. Method of Providing Digital Signatures. U.S. Patent 4,309,569, 5 January 1982. Available online: https://patents.google.com/patent/US4309569A/en (accessed on 15 August 2022).
[6] Wang, X.; Yu, H. How to break MD5 and other hash functions. Germany: Springer, 2005.
[7] NIST. Secure Hash Standard (SHS). Federal Information Processing Standard Publication. FIPS 180-4. 2015. Available online: https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180- 4.pdf (accessed on 15 August 2021).
[8] Wang, X.; Yin, Y.L.; Yu, H. Finding collisions in the full SHA-1. German: Springer, 2005.
[9] Dobbertin, H.; Bosselaers, A.; Preneel, B. RIPEMD-160: A strengthened version of RIPEMD. In Fast Software Encryption; Germany: Springer, 1996.
[10] Dwork, C.; Naor, M. Pricing via Processing or Combatting Junk Mail. In Advances in Cryptology-Proceedings of Crypto'92 Germany: Springer, 1993.
[11] Jakobsson, M.; Juels, A. Proofs of Work and Bread Pudding Protocols (Extended Abstract). In Secure Information Networks, USA: Springer, 1999.