International Journal of Recent Advances in Multidisciplinary Topics
Volume 2, Issue 11, November 2021
https://www.ijramt.com | ISSN (Online): 2582-7839

Understanding Crypto Currency from a Mathematical Perspective

Vedant Sharma*

***Abstract*: Bitcoin is a decentralized monetary system. This currency does not include physical notes or coins; it may only be used to make online purchases. A significant advantage of Bitcoin is that it is available worldwide and is not controlled by any single government or organization. Due to the fact that so many businesses operate online and across national lines, this might be quite useful. When businesses or individuals trade currencies, they want to avoid paying exchange fees and taxes. They may circumvent all of these costs by utilizing a digital currency such as Bitcoin. One of the most significant drawbacks of Bitcoin is the scarcity of retailers who accept it, but the Isle of Man has recently welcomed a rising number of businesses who do. Another issue with digital currencies is their youth; as a result, their value is vulnerable to extreme volatility. Additional concerns, such as hackers stealing Bitcoins, may occur. The Bitcoin protocol may be described as a mathematical method on a network that maintains transaction data and establishes majority consensus among the users. Thus, If the majority of people are being truthful, we'll receive an accurate result. This can be considered as a sincere and unforced agreement. The key characteristic of this system is that it is decentralized. This translates to suggest that there isn't a single centralized structure in command. The network's nodes are self-selected volunteers that are equal in terms of rights and responsibilities. The network is available to everyone, and everyone is welcome to take part. Having launched the network, it's incredibly durable and is considered to be invincible. It's been working fine for a long time, and since January 2009, there has been little to no disruption. The source code is available, as is the development process. The exact identical piece of code has hundreds of different cryptocurrencies have been created using the same technology that's been recycled and changed on the same foundation the protection of the network is protected by sophisticated cryptography much more secure than conventional cryptography's usage of monetary products and services. Classical hash functions, for instance digital signatures using elliptic curves and SHA256 ECDSA is being used as the search algorithm. It was encrypted using such methods which are quite well known so we won't go into detail about it because it's so commonplace. The cryptography tool's mathematics is quite complex, but its intriguing special characteristics drive cryptography research or alternative crypto assets. This paper seeks to explore the concept, i.e., Cryptocurrency from the perspective of Mathematics with bitcoin as its focal point.**

***Keywords*: Mathematics, Bitcoin, Cryptocurrency, elliptic curves, algorithms, cryptography, special functions, digital keys.**

1. Introduction

Crypto currencies are decentralized digital currencies that rely on cryptographic techniques to manage money production and fund transfers without the intervention of a trusted third party. The currencies are built on top of blockchain technology, which is a decentralized public ledger that can be programmed to record digital data. Bitcoin, the first and most popular cryptocurrency, was suggested in 2008 by an unidentified developer(s) Satoshi Nakamoto in the whitepaper 'Bitcoin: A Peer-to-Peer Electronic Cash System'. Bitcoin is a decentralized network of computers called nodes that are connected over the internet. [1] By running Bitcoin software on a computer with an internet connection, anyone with a computer and an internet connection may join the network. Each node on the network maintains an identical copy of a distributed ledger; a database storing the history of all network transactions.[1] Additionally, the blockchain is referred to as this. Due to the fact that anybody may join the network, nodes cannot completely trust one another. To address this, every transaction history is made public, depending on all nodes to agree on a single truth. This does not jeopardize the user's anonymity, however, because the database contains no personal information about the user, simply his or her public Bitcoin address and transaction history. [1] Additionally, users may generate several Bitcoin addresses using a single combination of public/private keys, enhancing privacy even further. We refer to the network as decentralized since there is no central authority verifying the authenticity of transactions; instead, proof-of-work is used to determine their validity. This boosts security because, in contrast to third parties (e.g. banks), if a network is attacked, just one node is impacted rather than the entire system. Bitcoin circumvents this issue by utilizing something called a "blockchain." This is a public record of all transactions to date. The blockchain records when Bitcoins are produced and when they are transferred between users [1]. Bitcoin has a really sophisticated mechanism for ensuring the blockchain's accuracy. If the blockchain could be manipulated, it would be easy to steal or copy the currency, and Bitcoin would cease to function, as it requires an exact record of all transactions to function. Computers that tackle massive number crunching problems verify the blockchain's correctness. Bitcoin is based on hundreds of computers worldwide solving complex tasks in order to validate transactions and establish the blockchain's accuracy. Computing power is not a free resource. Calculations demand extremely powerful computers, which are expensive, as well as

*Corresponding author: vedantsharma1220@gmail.com

storage space and electricity to run [2].

2. How do these Mathematical Functions Work

1) Nodes

The bitcoin network is comprised of nodes, or machines that run the bitcoin Programme and interact with one another. Nodes that comply to a stringent set of standards broadcast and validate bitcoin transactions as they traverse the network. You cannot compel nodes to follow these rules. When incentives make disobeying the laws economically expensive, a virtuous loop is created [3]. As a result, the network is a dynamic, extremely intricate system with no guarantee of stability. The mathematical issue of system stability is extremely fascinating and significant. In this paper we will try to decipher the following terms: special functions, martingale theory, Markov chains, and Dyck words, among other subjects [1].

2) Mining

The network's nodes broadcast transactions, and any node can join. Mining is a word that refers to the process of transaction verification, which is also associated with the generation of new bitcoins.[4] The protocol rules determine the rate at which new bitcoins are produced as a result of the "electronic gold" analogy. Coinbase transactions generate additional bitcoins without the user's input. These transactions take place on a consistent schedule, with one block of transactions verifying every ten minutes [6].

3) Transactions

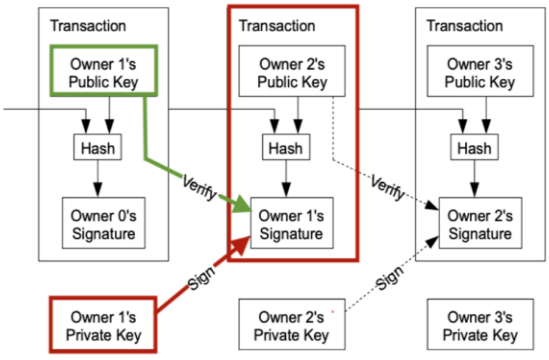

Fig. 1. Transactions of Bitcoin - Bitcoin Whitepaper 2008

A transaction is when you transmit or receive Bitcoin from one location to another. Each larger box in the accompanying illustration represents a distinct transaction inside the transactional process. [2]To create a new transaction, a hash of the previous transaction inputs (transactions in which the user received their Bitcoin) and the payee's public key (Owner 2) is utilized. The sender (Owner 1) signs the transaction with their private key to prove that it originated with them, therefore authorizing it. By authenticating the signature with the sender's public key, the payee may validate that the sender is authorised to spend the transaction's value (highlighted in green). [1] A public record of all transactions enables us to search for transactional data such as the one below. Each transaction is allocated a unique transaction value in the form of a 256-bit hash, which is used to uniquely identify and validate it. This is shown at the screen's top [4]. The hash value for the transaction code is determined using the SHA-256 algorithm.

4) Blocks [2]

Before a new block is added to the network, it must include at least one transaction. A block is a collection of transactions. Furthermore, block data is easily accessible to the general public via the internet. Additionally, the block contains a unique 256-bit hash value that is used to identify and validate the block, same to how transaction data is identified and validated. This value is obtained by the SHA-256 algorithm being applied to the transaction.

5) Understanding the Blockchain [1]

Each of these blocks is cryptographically linked together to form the "blockchain," which is a chronological record of all bitcoin transactions to date. The current blockchain has around 260.000Mb of data. The mining/validation technique creates a cryptographic link between blocks by employing a hash function and a "Proof of Work." [3] A tremendous amount of computer power is required to validate a block, which is why the data cannot be edited or damaged. To change a single bit in a block, we must first do all of the calculations needed to generate the preceding blocks. At the moment, no government or organization has the computing capacity to make even the smallest alteration to the blockchain's more than 600,000 blocks [4].

Decentralized lotteries are used for mining and validation. A miner (a transaction validation node) creates a header for a new block by merging previously unvalidated transactions into a single block and hashing the previous block's header. The SHA-256 hash process is done twice, yielding a 256-bit result. Hash functions are straightforward to implement, but it is nearly hard to detect previous images or collisions (two files giving the same output). [8] It also has pseudo-random properties, which means that when a bit of the input is modified, all subsequent bits of output behave as uncorrelated random variables with values ranging from 0 to 1.[11]To begin mining, discover a hash number that is less than the difficulty, which is a preset value that must be met. the impediment The site is updated every two weeks to ensure that the validation rate remains constant at one block every year (2016 blocks), or approximately once every ten minutes. The pseudo-random features of the hash function ensure that this hash can only be determined by probing the system. Many hashes are affected when a header parameter is changed. The block is created by the first miner who solves the challenge. The block is then made public, and the network uses it as the last block in the distributed ledger technology known as blockchain.[12] Two blocks can be validated concurrently at various points around the network. Following that, a competition between two participants will be held, with the winner being the one who has a mined block on top of the network at the end of the tournament. The block that belonged to the other person gets discarded. This is referred to as a "orphan block." By storing currency on a blockchain, the acceptance rate of a block as part of a node is dependent on its ability to perform more work [1]. When the last block of a transaction contains a transaction, we refer to the blockchain as having one confirmation. Any more blocks produced by this

mining operation will be added to the existing Confirmation. This method records the transaction and adds it to the blockchain. This onerous technique is required to ensure that the network or blockchain cannot be manipulated. To have a vote in the validation decision, each member must make some kind of computational commitment. Essentially, building a decentralized money system was difficult since eliminating duplicate expenditure without the help of a central accounting agency was impossible. As a result, evaluating the likelihood of meeting the double spend problem was Nakamoto's first mathematical challenge [1].

Additionally, Satoshi designed the protocol's incentive for fast propagation.[1] The original block reward was 50 BTC, and it is half every 210,000 blocks thereafter. The Bitcoin supply formula shows the method for calculating the total number of Bitcoins mined during each halving period with? = the term of rewards. After adding the 0th and 32nd periods, we have 21 million Bitcoins [1].

$$\frac{\sum_{i=0}^{32} 210000 \times \left[\frac{50 \times 10^8}{2^i}\right]}{10^8} \approx 21{,}000{,}000$$

Fig. 2. [2]

As shown in Fig 2 There are 21 million Bitcoins in circulation in total. If this logic remains true, there will never be more than 21 million Bitcoins in existence. Satoshi never said precisely why the 21 million limit was selected, but Schellinger believes it was a combination of simplicity and processing efficiency. According to Schellinger,[2] monetary policy was devised first since it is what mattered. Every four years, mining incentives are halved, resulting in the generation of half as many bitcoins and halves the number of bitcoins created in the prior period [2]. Given the rapid rate of block solving, it is projected that all Bitcoins will be mined by 2140. [3] When all coins have been produced, miners will earn cash by increasing transaction fees. When seen by the graph above, the inflation rate decreases as block rewards are half every 210 (thousand) blocks. According to the Bitcoin code, there are only 21 million Bitcoins in circulation. As the total supply of Bitcoin approaches zero, the inflation rate will decrease to zero as the currency's value grows over time [3]. As previously stated, the information above is concatenated to provide a unique transaction block identity. These are the following elements: version number, hash of the preceding block, Merkle root, timestamped hash, difficulty, and nonce. The block header is a collection of six of these components. When the hash of a transaction block is discovered, it is really the hash of the block header. The hash of this block serves as its single identifier.

6) Version Number

This block was constructed using version number, 2, of the Bitcoin core utility. All Bitcoin network users now have access to the rules that were in effect at the time this block was generated and hashed. [4] If you use an older version of Bitcoin core, the block chain may fork. The hash of the previous block is the result of hashing the header of the transaction block [4].

7) Merkle Root

HA denotes the hash of transaction A for the purpose of computing the Merkle root. The Merkle root is constructed from the Merkle tree's 'leaves' and reflects the total of all transactions included inside this transaction block. [9] Transactions are used to depict the Merkle tree's leaves. To determine the block's root, a double-SHA256 algorithm is used to hash all of the transactions in the new block. This operation is continued until there is only one remaining 256-bit hash, the Merkle root. Each hash pair's result is then paired and hashed, and so on. The Merkle root is used to produce a transaction identity that is one-of-a-kind. Once all transactions are double-hashed, any effort to modify that information results in a complete revision of the block's root, hence strengthening the block's security. The inclusion of a transaction in a transaction block may be established by establishing that the transaction exists in the Merkle tree for that block.

As a result, a node would gather the records of all other transactions in the block, hash them, and then compare the hash to the Merkle root contained in the block header to determine whether it is genuine [7] Therefore, to verify that transaction J is included in this block, get or recreate the hashes ABCDEFGH, MNOP, KL, and I from other nodes and complete the Merkle tree by comparing this current hash to the Merkle root. These two transactions are compared to determine if they are consistent, and if they are, the proposed transaction J is added to this transaction block.

3. Time Stamped Hash

The timestamp on the block indicates when it was created. A timestamp enables you to establish precisely when each transaction in a block first appeared in the ledger. In the future, no transactions with the same bitcoins as those in this block will be allowed. To arrive at this result, the transaction identifier is compared to those in the transaction pool and those in the block chain. Bear in mind that when a bitcoin input is transformed to an output, the identifier linked with the bitcoin input changes. Only the output of the block may be used as an input to another block [7]. The timestamp is critical to avoiding unintentionally spending the same bitcoin several times. When an unspent transaction output is used as an input in a future transaction, the timestamp verifies the transaction occurred and prevents the same bitcoin or transaction from being included in a subsequent timestamped block (Nakamoto 2008).

1) Difficulty

To assess the difficulty, you must first identify the difficulty of your present aim. Miners would have a very easy time obtaining a successful block, as the difficulty would be set at one. For instance, when block #350650 was mined at a difficulty of 1, the difficulty was set to 1 and the difficulty value was 46,717,549,644.71. The difficulty is also considered while establishing the goal or maximum hash value [7] later on, we'll discuss the purpose and complexity of proof-of-work.

2) Nonce

A random number is included in the block header as the nonce. Miners will try various random values in the header until the hash of the header fulfils Bitcoin core's proof-of-work

requirement. Once a nonce is discovered, miners can modify the timestamp to extend it by up to two hours beyond actual time or the coinbase transaction to begin searching for another nonce [7].

3) Proof of work

To be added to the blockchain, the mining node that generated the block must first demonstrate that it solved a computationally difficult problem. The Merkle root and preceding blocks' SHA-256 hashes yield a value less than a predefined value specified by the current difficulty, which the miner alternates between using and not utilizing (beginning with a certain number of zeros).[7]

4. The Problem of Double Spend [1]

The current transaction technique has a problem with 'double-spending.' Double-spending is the act of spending the same amount of money twice. If you're using regular currency, this is irrelevant because you can either give the receiver cash or have a third party (such as a bank) keep track of how much money has been spent previously. We must have a mechanism in place to prevent someone from broadcasting another transaction using the same money before the previous one has been completed, given Bitcoin transactions are slow to execute and uncontrolled. Time stamping serves as a safeguard against this. Time stamping establishes the chronological order of blocks by integrating the time the block was added and the timestamp of the preceding block into the block's hash value. Thus, the hash cannot have been updated after it was formed, as the subsequent block would have a completely different hash. The order of a transaction is immutably stored in each block's timestamp, and this record gets increasingly secure with each consecutive block. Double-spending is conceivable only if an attacker has access to more than half of the total computing power on the Bitcoin network. Even with a 51 percent advantage, it is improbable that the assailant could solve all of the blocks in a row faster than the other 49 percent. As a result, a user cannot execute two transactions that use the same input simultaneously while the first transaction is being completed. Initially, the first block gets verified as part of the blockchain with the next block containing data from the previous block. If the input from the previous block has already been utilized, the next block then proceeds to reject the transaction. (Fig 2)

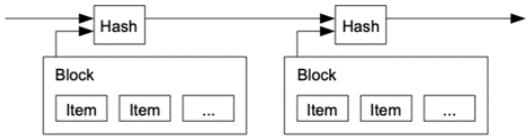

Fig. 3. Bitcoin Whitepaper, 2008

This can be accomplished only by rewriting the blockchain beginning with the first transactional block that has been validated on the official blockchain and the seller has delivered the things (the vendor will not deliver until several confirmations are shown). [8] He has the ability to alter the blockchain's ultimate conclusion as long as he controls a majority of the network's hash rate, which implies his relative hash rate exceeds half of the network's hash rate. This is to ensure that no single entity controls more than half of the available mining capacity. Nonetheless, even when the probability function is equal to zero, he can attempt and succeed at a double spend. [4] The probability that the rogue miner modifies the most recent n 1 blocks is the first mathematical issue. p = 1/q is the remaining relative hash rate, which we assume is composed of ethical miners who follow the protocol's rules.\

This problem is similar to the classical gambler's ruin problem. Nakamoto observes that the probability of catching up n blocks is

$$q_n = \left(\frac{q}{p}\right)^n \qquad \text{(Nakamoto)}$$

[1]

Miner behaviour can be modelled by looking at how long it takes for processes like N(t) and N(t) to complete counting the number of mined blocks at time t for both honest and malevolent miners.

$$p = \frac{\alpha}{\alpha + \alpha'}, \quad q = \frac{\alpha'}{\alpha + \alpha'}.$$

[1]

When the honest miners mine their nth block, the attacker's random variable Xn = N(Sn) is a negative binomial variable with parameters (n, p). As a consequence, when an integer k 1 is supplied, we obtain

$$\mathbb{P}[\mathbf{X}_n = k] = p^k q^k \binom{k+n-1}{k}.$$

[1]

The traditional approximation of a negative binomial variable by a Poisson variable implies that he studies it. According to Rosenfeld [5], a more accurate approximation is the negative binomial variable. We could obtain the exact formula for the double spend likelihood after z confirmations.

Theorem 1 ([3], 2017). *After z confirmations, the probability of success of a double spend by attackers with a relative hashrate of* $0 < q < 1/2$ *is*

$$P(z) = I_{4pq}(z, 1/2)$$

where $I_x(a, b)$ *is the incomplete regularised beta function*

$$I_x(a, b) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \int_0^x t^{a-1}(1-t)^{b-1}\, dt\,.$$

This probability computation is crucial for Bitcoin's safety. It's a real-world finding, not simply a hypothesis. A transaction's estimated risk and the number of confirmations needed to deem it final can both be reversed using this method. After six confirmations, the likelihood of a double spend, for example, is less than one percent if q = 0.1 [1]. This probability computation is crucial for Bitcoin's safety. It's a real-world finding, not simply a hypothesis. A transaction's estimated risk and the number of confirmations needed to deem it final can both be reversed using this method. After six confirmations, the

likelihood of a double spend, for example, is less than one percent if q = 0.1. [1]

Nakamoto created a computer simulation to figure out how likely it was that his argument would be correct. According to his reasoning, the chance drops off exponentially with increasing number of confirmations (as stated by him, we may witness a decline in the probability exponentially with z).[1] Even though the numerical simulation doesn't prove anything, this claim has been made often, but it was never proven until 2017. We can show the following Corollary with the prior precise formula and traditional methods: [9]

Corollary 2. *Let $s = 4pq < 1$. When $z \to +\infty$ the probability $P(z)$ decays exponentially, and, more precisely,*

$$P(z) \sim \frac{s^z}{\sqrt{\pi(1-s)z}}.$$

[1]

5. Public Keys

Public key cryptography protects data by encrypting it with a pair of keys: one public and one private. Encrypting keys, sometimes referred to as public keys, can be made publicly available. Finding the original communication is far more difficult without the decrypting key, which is significantly more difficult than using the public key for the first encryption. If you're utilizing a public key cryptosystem, user A chooses a public key E and makes it publicly available, while keeping his private key and the process by which the public key was generated secret, in accordance with an agreed-upon set of rules and regulations.

A will use his/her public key to encrypt each message m it receives, culminating in s = E. (m). To decode m, A must first receive the encrypted message s and decrypt it using his private key D. [3]. Public keys can also be used to produce digital signatures. If a transmission is genuine, digital signatures can verify that it was transmitted by the intended recipient and that no data was altered after it was received, therefore establishing its validity. Signatures are generated by fusing the private key with a modified version of the original message. The public key may be used to authenticate the message's authenticity by decrypting the message's signature using D(E(m)) [3].

When bitcoins are transferred between wallets on the Bitcoin network, cryptographic signatures and keys are used to establish ownership. When a user creates a new wallet in the wallet application, a private key is produced. While this private key is kept secret, it is used to produce all public keys. In Bitcoin, the wallet address is the hash of the user's public key [7] Once a user spends 19 bitcoins, the previously unspent bitcoins are signed with the user's digital signature. This signature ensures that no transaction data has been altered since the bitcoin was received, and it also verifies that the user has the authorization to spend this bitcoin. Bitcoin's architects faced a dilemma when it came to selecting a public key cryptography scheme.

1) Elliptic Curve Cryptography

While it comes to providing security, elliptic curve cryptography takes little computational power, but when trying to crack it, it necessitates a significant amount of computing power [11]. Rosen [3] says that "the set of points (x,y) that fulfil y2=x3+ax+b where a and b are real values" is an elliptic curve. [3] x and y are actual numbers in the following example. Existing points on the curve can be used to generate new ones. By adding P and Q to R at two places on the curve, one can get to R at those same spots. There are two main ways to go about it [3]. Let P=(x1,y1) and Q=(x2,y2) be the first approach, with PQ. Draw a line across P and Q to get to R, then intersect the curve at R′ to get to R. Adding points P and Q gives us R, which is the reflected value of R′ and which equals (x3,y3) on the x-axis (Fig. 3). This outcome can also be discovered using algebra [3]

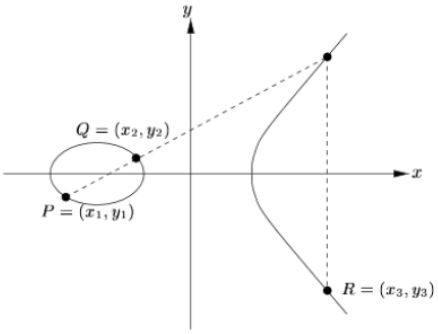

Fig. 4. Point Addition [7]

To find R, first find the slope of line l given by $m = \frac{y_2 - y_1}{x_2 - x_1}$, and the equation of the line given by $y = m(x - x_1) + y_1$. Hence, by substitution:

$$(m(x - x_1) + y_1)^2 = x^3 + ax + b$$

$$y_1^2 + 2my_1(x - x_1) = x^3 - m^2(x - x_1)^2 + ax + b$$

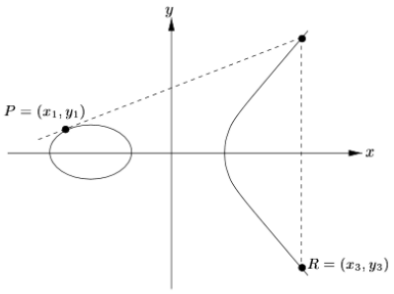

Fig. 5. Point Doubling [12]

Since the sum of a cubic function's roots equals the negative of the squared term's coefficient, we know x1+x2+x3=m2, and x3=m2x1x2=x3. If we want to know the value of y3, we may solve for it as follows: y3=m(x3x1)+y1 and get R′=x3,y3 [12] P=Q, PQ is the point doubling formula for the second approach. Note that when Q approaches P, the slope of the line passing between P and Q approaches the slope of the tangent line to P. [4] This will help you determine the line l. Once P=Q, 2P=R, and R=(x3,y3) is the reflection of R′ over the x-axis, the equation becomes (Fig. 3.1). The slope of line l is provided by m=(3x12+a)2y1, which is derived by implicitly differentiating

the curve at P. Similar methods are used to locate R's coordinates to those already mentioned: x3=m22x1 and y3=m(x1x3)y1 [3]. As of right now, this public key does not have a corresponding wallet address. A=RIPEMD160(SHA256(K)) is the hash used to obtain the wallet address by concatenating the public key K's coordinates with the RACE Integrity Primitives Evaluation Message Digest (RIPEMD) digest from the SHA256 hash. This generates an A number of 160 bits [7].

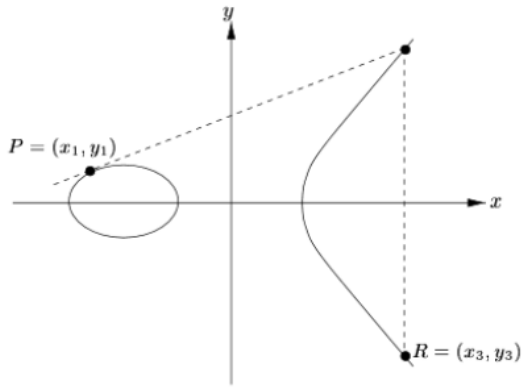

Fig. 6. 2 y2 = x2 + 7 [1] Fig 3.3 K = kG[3]

2) SHA-256

Cryptographic hash functions take a string as input and output an alphanumeric value as an alphanumeric value of a predetermined length. Bitcoin uses a hash algorithm called SHA-256 (Secure Hashing Algorithm 266) to generate a message digest from an input message. There is a restriction of 2256-1 bits on the length of the data, therefore the output will always be 256 bits long. Because it's a one-way operation, hashing can't be used as a kind of encryption or decryption (decrypted). We can examine how the result changes even when only one character in the message is changed by passing numerous outputs through SHA-256. We can also observe that despite the lengthier input, the output is exactly the same in length. Additionally, SHA-256 is deterministic, meaning the output will always be the same when given the same input.[12]

The National Security Agency created this method as a variation of SHA-2 (Secure Hash Algorithm 2). (NSA). [12] SSL, TLS, SSH, and Unix/Linux all employ SHA-256, as do many other prominent encryption protocols and operating systems [10]. As previously stated, the hash algorithm is incredibly safe and does not have its inner workings divulged to the general public. U.S. government agencies rely on it to safeguard confidential data because of its capacity to validate data content without disclosing it owing to digital signatures. Because it does not need the storing of specific passwords, hash values can be maintained and compared to user entries in order to check if they are accurate or not. Furthermore, it is also used for password verification. Actually, a hash value can't be used to decode the original data because of this. In addition, the sheer number of possible combinations makes a brute force approach very impossible. In addition, the likelihood of two data values (known as collisions) having the same hash is extremely low.

6. Profitability of Crypto Currency Mining

The protocol's stability is a crucial issue to consider after examining its security. Individuals' interests must be appropriately matched with the protocol rules in order for a decentralized protocol to function correctly.... Miners should get the most possible profit by strictly adhering to the protocol's requirements. As we've learned from studying unstable dynamical systems, this isn't an easy feat. It's a little surprise that this has been empirically proved from Bitcoin's beginning. For example, it's far from evident that publishing a block as soon as a miner validates it is in the miner's best interest. However, he risks the danger of another miner publishing a confirmed block and therefore adopting it on the public blockchain, resulting in him losing his reward. He can keep it hidden and discreetly push his advantage. In the Bitcoin talk forum, which Nakamoto started in 2010, this kind of scenario has been explored since 2012. In order to answer this, we must first create a realistic profitability model. Every firm measures its "Profit and Loss" per unit of time, including the mining industry. A miner's earnings are derived from the transaction fees and block rewards, which include the Coinbase reward in new bitcoins generated. When t > 0, profitability is given by

$$\mathbf{PL}(t) = \frac{\mathbf{R}(t) - \mathbf{C}(t)}{t}$$

where $\mathbf{R}(t)$ and $\mathbf{C}(t)$ represent, respectively, the rewards and the cost of the mining operation up to time t. If we don't consider transaction fees, we have

$$\mathbf{R}(t) = \mathbf{N}(t)\,b$$

Coinbase rewards are calculated as follows: b > 0. Even if transaction costs are included, the last equation still holds when using the conventional Wald theory to calculate the average reward [6]. Due to external factors (such as power prices, mine hardware costs, geographic location, currency exchange rates, etc.), the random variable C(t) indicating mining operations' cost is significantly more difficult to ascertain. However, as we'll see in the next section, it's not necessary when comparing the profitability of various mining tactics [1]. The mining process is repeated, with miners returning to the same starting point to begin mining a new block after a period of time. An effective mining plan is made up of cycles that always go back to the start. It's a "repetition game" like the ones used by successful casino gamblers (when they find a flaw in the game that makes it profitable). When a new block is validated by the network, an honest miner will begin a new cycle [5]. In the current form, the double spend technique is flawed since it has a non-zero failure probability, and if we keep mining in the hopes of catching up to the official blockchain from a significant distance behind, we have a positive chance of ultimate devastation. [5] In addition, because the attack is intended to last infinity, the approach cannot be integrated. As a result, we must set a bar for ourselves because we are so far behind the official blockchain [5]. Since the beneficiary of the transaction has requested z confirmations, we can safely believe that we will never fall behind a z official blockchain blocks. The A-Nakamoto double spend approach is one such integrable

strategy. After z confirmations, the probability of success of an A-Nakamoto double spend is [1].

$$P_A(z) = \frac{P(z) - \lambda^A}{1 - \lambda^A}$$

where $P(z)$ is the probability from Theorem 1 and $\lambda = q/p$.

If v is the amount to double spend, then we can compute the revenue ratio $\Gamma_A = \mathbb{E}[\mathbf{R}]/\mathbb{E}[\tau]$.

Based on the foregoing, the A-Nakamoto double spend strategy's predicted revenue and duration are

$$\mathbb{E}[\mathbf{R}_A]/b = \frac{qz}{2p} I_{4pq}(z, 1/2) - \frac{A\lambda^A}{p(1-\lambda)^3[A]^2} I_{(p-q)^2}(1/2, z) + \frac{2 - \lambda + \lambda^{A+1}}{(1-\lambda)^2[A]} \frac{p^{z-1}q^z}{B(z,z)} + P_A(z)\left(\frac{v}{b} + 1\right)$$

$$\mathbb{E}[\mathbf{T}_A]/\tau_0 = \frac{z}{2p} I_{4pq}(z, 1/2) + \frac{A}{p(1-\lambda)^2[A]} I_{(p-q)^2}(1/2, z) - \frac{p^{z-1}q^z}{p(1-\lambda)\,B(z,z)} + \frac{1}{q}$$

with the notation $[n] = \frac{1-\lambda^n}{1-\lambda}$ for an integer $n \geq 0$, and B is the classical Beta function.

In such a situation, a business will be undermined by participating in a significant double expenditure. The amount from which a double spend is lucrative for a tiny miner with relative hash rate 0 < q << 1 may be estimated.

Corollary 14. *When $q \to 0$, the minimal amount v for an Nakamoto double spend with $z \geq 1$ confirmations is*

$$v \geq \frac{q^{-z}}{2\binom{2z-1}{z}} b = v_0 .$$

[9]

7. Conclusion

Due to its ability to alter our society, Bitcoin serves as a good illustration of the universality of mathematical applications.

References

[1] Nakamoto, Satoshi. "Bitcoin: A Peer-to-Peer Electronic Cash System." Satoshi Nakamoto Institute. October 31, 2008.

[2] H. Massias, X.S. Avila, and J.-J. Quisquater, "Design of a secure timestamping service with minimal trust requirements," In 20th Symposium on Information Theory in the Benelux, May 1999

[3] Rosen, Kenneth H. "Cryptology." In Elementary Number Theory and Its Applications. 6th ed. Reading, Massachusetts: Addison-Wesley Pub., 2011

[4] Chitty, Toby. “The Mathematics of Bitcoin-Digital Keys.” Medium. Medium, May 28, 2020.

[5] Perez-Marco, Ricardo. "Bitcoin and decentralized trust protocols." arXiv preprint arXiv:1601.05254 (2016).

[6] Rosenfeld, Meni. "Analysis of hashrate-based double spending." arXiv preprint arXiv:1402.2009 (2014).

[7] Johar, Sumaira, Naveed Ahmad, Warda Asher, Haitham Cruickshank, and Amad Durrani.. "Research and Applied Perspective to Blockchain Technology: A Comprehensive Survey" *Applied Sciences* vol. 11, no. 14, pp. 6252, 2021.

[8] Antonopoulos, Andreas M. Mastering Bitcoin. Sebastopol, CA: O'Reilly Media Inc., 2015.

[9] Prat, Julien, and Benjamin Walter. "An equilibrium model of the market for bitcoin mining." 2018.

[10] Grunspan, Cyril, and Ricardo Pérez-Marco. "On profitability of selfish mining." arXiv preprint arXiv:1805.08281 2018.

[11] Grunspan, Cyril, and Ricardo Pérez-Marco. "On profitability of stubborn mining." arXiv preprint arXiv:1808.01041 2018