Production Systems and Information Engineering
Volume 11 (3), pp. 154–172.
https://doi.org/10.32968/psaie.2023.3.7

BLOCKCHAIN AND HASHING ALGORITHMS: A REVIEW

Yassir Soulaimani
University of Miskolc
Hungary Institute of Information Technology
yassir.soulaimani@gmail.com

Károly Nehéz
University of Miskolc
Hungary Institute of Information Technology
aitnehez@uni-miskolc.hu

Abstract: Cryptocurrencies use a secure, distributed ledger system called blockchain and mining is an essential part of it. It adds records of past transactions, enables consensus, and creates new units of currency. They are designed as peer-to-peer systems and rely on miners to validate transactions. The paper evaluates different mining techniques used by major Cryptocurrencies, analyzing their strengths, weaknesses, and potential threats. It provides an overview of the various ways in which Cryptocurrencies can be mined and highlights their unique strengths and vulnerabilities.

Keywords: *Blockchain, Cryptocurrency, Mining*

1. Introduction

Cryptocurrency represents a groundbreaking digital exchange system that leverages the potential of cryptography to create and facilitate the distribution of digital currency units [1]. Unlike traditional financial systems, Cryptocurrencies operate in a decentralized fashion, eliminating the necessity for a central authority to oversee transactions. Instead, they rely on a distributed network of participants to verify and record transactions through a process called mining [2].

The emergence of Cryptocurrencies was driven by the shortcomings and inefficiencies of conventional financial systems. These digital currencies harness state-of-the-art technology to offer secure, transparent, and efficient methods of transferring value across the internet. At the heart of this technology lies the blockchain, a distributed ledger that securely and immutably records all transactions [16].

These Cryptocurrencies are not uniform; they differ not only in their underlying technologies but also in their primary goals. Some Cryptocurrencies prioritize scalability, striving to handle a high volume of transactions per unit of time. Others emphasize security and aim to deliver fast and lightweight services [3].

The wide array of mining algorithms and consensus mechanisms employed by various Cryptocurrencies reflects their distinct objectives and characteristics.

In this paper, we embark on a journey into the captivating realm of Cryptocurrency mining systems, subjecting them to an efficiency evaluation. Our analysis encompasses a carefully selected group of prominent Cryptocurrencies, including Bitcoin [5], Litecoin [6], Peercoin [7], Ethereum [8], Ripple [9], Namecoin [10], Auroracoin [11], Blackcoin [12], Dash [13], Decred [14], and Permacoin [15]. These Cryptocurrencies have not only achieved broad adoption but also exhibit intriguing technological features, commanding the highest market capitalization and transaction rates in the Cryptocurrency landscape. Our exploration of these Cryptocurrencies aims to furnish readers with a comprehensive grasp of the predominant mining algorithms and consensus mechanisms currently in use.

The structure of this paper is meticulously designed to facilitate an understanding of Cryptocurrency technology. In Section II, we provide clear definitions of pertinent terms, ensuring comprehension of the concepts discussed throughout the paper. Section III offers a historical perspective and background, tracing the evolution of Cryptocurrencies from their inception to their contemporary status. Section IV delivers an exhaustive overview of blockchain technology [16], which underpins nearly all Cryptocurrencies. Section V introduces the fundamental principles of mining, while Section VI delves deeper into the specifics of Cryptocurrency mining systems. Section VII delves into the pivotal role of hash algorithms in safeguarding the security of Cryptocurrencies. Section VIII tackles some of the primary challenges and issues associated with Cryptocurrencies, offering insights into potential solutions. Lastly, in Section IX, we conclude our analysis, summarizing key findings and contemplating the prospects of the Cryptocurrency landscape.

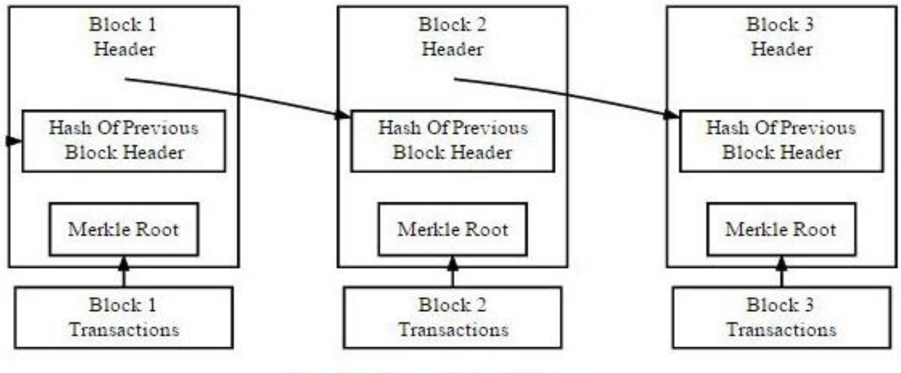
Through this paper, we aim to shed light on the intricate world of Cryptocurrencies and mining systems, providing valuable insights into the technology that has disrupted traditional finance and continues to reshape the global financial landscape.

2. History and general working principles of Cryptocurrencies

The first fully implemented decentralized Cryptocurrency was Bitcoin, published by Nakamoto in 2008–09 [5]. Before this, there were published articles about peer-to-peer currency systems, but none were implemented. Following the success of Bitcoin, several others came into existence [19].

Chaum created an anonymous electronic money system called eCash in 1983 [20]. The main difference between eCash and Cryptocurrencies is that eCash was centralized (via banks). Software on the user’s local computer stored money digitally, which was cryptographically signed by a bank [20].

PayPal is an online money transfer system established in 1998 [21]. PayPal provides users with an account, which can be linked with bank accounts and credit cards, and users can pay someone or receive payment through PayPal accounts. PayPal does not have its currency.



Simplified Bitcoin Block Chain

Figure 1. *A Bitcoin Blockchain* (adopted from [5])

M-Pesa [22] was established by Vodafone initially in Africa, which later spread to other continents. M-Pesa is a mobile, online payment system in which users can deposit money into an account stored in their cell phones and send PIN-secured SMS texts to other users to send money [22]. All these online monetary systems were based on fiat currencies [23], whereas a Cryptocurrency has its own currency.

Cryptocurrencies work functionally as follows [19]:

- The user has a wallet with a generated address. This address acts as a public key [24].
- The wallet also contains a generated private key, which is used to sign transactions, proving ownership [24].
- The payer sends money to the payee's address and signs it is using the payer's private key.
- The transaction is verified by mining [2].

3. Blockchain Overview

A Blockchain is a public ledger of Cryptocurrency transactions that is distributed across a network [17]. Each block [25] in the Blockchain contains a certain number of verified transactions. The maximum number of transactions that can be included in a block is set by the Cryptocurrency system. For example, the maximum size of a Bitcoin block [5] is 1 MB. Figure 1 shows a simplified representation of a Bitcoin Blockchain. A Bitcoin block is composed of five fields [25]:

- **Magic number** – which is fixed
- **Block size**
- **Block Header** – which contains the hash of the previous block, the time stamp, the block version number, the hash based on all the transactions in the block, and the nonce.

- **Transaction counter** – which is the number of transactions included in the block.
- **Transactions** – the enumerated set of verified transactions added by the block.

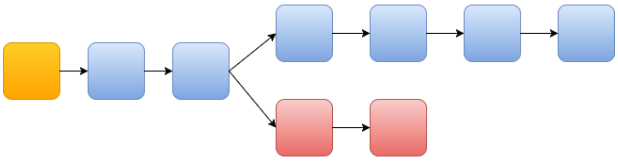


Figure 2. *Forking in a Blockchain* (adopted from [17])

The first block ("genesis block") contains the first transactions of a given Cryptocurrency. The hash of the first block is passed forward to the miner, which uses it and generates a nonce to create a hash for the second block. Likewise, each block contains the hash of the previous block, creating a chain from the first block to the current block through the inclusion of hashes. This creates a unique path from the most recent block to the first block, making it challenging for an attacker to tamper with information in a block because all subsequent blocks would need to be regenerated, which would be detected. The final hash would not match [17].

When two blocks are created at almost the same time, a fork occurs. The block created first, according to the timestamp in the block header is accepted in the chain, and subsequent blocks link to the accepted block. Figure 2 illustrates this.

4. Mining Overview

Every Cryptocurrency system studied incorporates a distributed public ledger called the Blockchain [16]. A transaction is created when a payer sends some currency to a payee. Mining validates these transactions and adds them to this public ledger. When a new transaction takes place, the miner checks if the currency belongs to the payer, or if the payer is trying to double spend [26]. The ownership of the currency is available in Blockchain.

To prevent malicious users from creating multiple nodes and trying to validate an invalid transaction, miners are required to solve a resource-intensive task. This makes it expensive for a malicious user to create enough false identities to outnumber benign users and validate an invalid transaction.

The resource-intensive task can be any of the following:

- *Proof of Work [27], which is a verifiable result of a resource-intensive task that confirms that the task has been performed.*
- *Proof of Stake [7], which requires the miner to prove how much currency they own in the system.*
- *Proof of Retrievability [15], which requires the miner to demonstrate that the data they were given to store is intact and can be retrieved at will.*

We are not aware of other proof methods at present. Details of these Proof systems are discussed below in section 6.

Proof construction requires intensive use of memory and/or computational power. The proof Requirements also restrict the number of transactions that can be validated (and consequently the number of blocks added to the ledger) in each period. This restriction is necessary because, with each block mined, new currency units—the total of which is finite are produced.

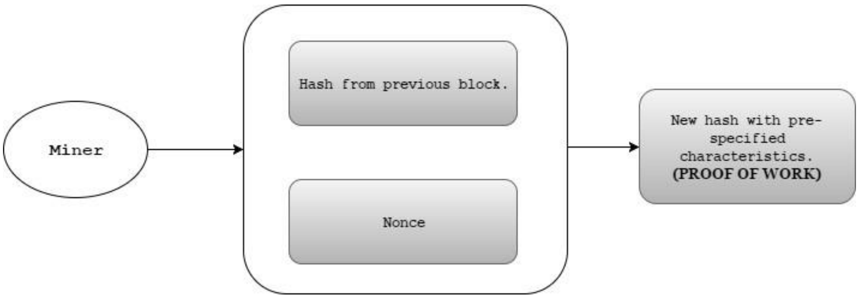

Miner
Hash from previous block.
Nonce
New hash with pre-specified characteristics.
(PROOF OF WORK)

Figure 3. *A simplified view of Proof of Work mining* (adapted from [2])

To prevent untimely exhaustion of Cryptocurrency, it's necessary to slow down the rate of production. For example, in the case of Bitcoin, every block currently introduces 50 new Bitcoins in the system. The number of new Bitcoins introduced is halved every 210,000 blocks. Consequently, through simple geometric progression, there can be at most 21 million Bitcoins [28]. If the number of Bitcoins mined per day wasn't restricted, the Bitcoin reserve would be exhausted far earlier than desired. After the limit is reached, the number of bits identifying a Bitcoin could be increased to create more units (this is a potential future event for Bitcoin).

<u>Here are the steps involved in mining:</u>

- A miner performs a resource-intensive task and produces proof that the work has been done [27]. This task prevents a malicious miner from forming false identities and manipulating them. Figure 3 offers a simplified view.
- The proof produced is verified to confirm that the task has been performed.
- The miner then checks the validity of the transactions, and if all the transactions in the block prove valid, the block is posted in the Blockchain [2].
- Requirements

Mining is a brute-force algorithm and should be designed so that the number of blocks mined per day remains approximately constant to control the rate of introduction of new currencies, which are unlocked when a block is mined [28]. The first miner to compute the proof gets to validate the block and earns the reward, which is a fraction of the unlocked currency. The Proof produced by the miner needs to be verified. This verification should be fast and easy.

Technological overview:

Cryptocurrencies usually mine with one-way functions (e.g., hashes) [2]. The miner gets the hash of the previous blocks as input. He/she must choose a nonce such that when the current hash and the nonce are hashed, the result follows a structure defined by the Cryptocurrency (e.g., Bitcoin requires that the output must have 0s in the N most significant bits [5]). Calculating an input from the hash is resource-intensive, whereas verifying its correctness by calculating the hash is fast. Hash functions are designed so that determining the input from the output is extremely time-consuming, making it intractable [18]. The miner must generate nonces and try hashing them with the given input until the requirements are fulfilled [2]. The computational complexity of the reverse hashing function is significantly higher than the hashing function since it is a brute-force algorithm. Finding the correct nonce is resource intensive as well as time-consuming since it involves calculating a large number of hashes, whereas verifying that the nonce, when added to the hash of the previous block, produces a new hash that fulfills the requirements is a matter of one hash computation and is fast [2]. Proof of Stake systems are usually not used independently, but rather are coupled with Proof of Work [29].

Controversies:

A major flaw in the proof of work system is the 51% attack [30]. If a single entity controls more than half of the total mining hash rate, then that entity would be able to manipulate the Blockchain at will. An attacker who controls more than 50% of the network's computing power can, for the time that they are in control, exclude and modify the ordering of transactions. This allows the successful attacker to perform the following operations [31]:

- Reverse transactions that they send.
- Prevent some or all transactions from validation.
- Prevent some or all other generators from getting any generations.

While this is theoretically possible, it would require the attacker to have access to immense resources. Acquiring such resources would be expensive and the overall expense might well exceed the potential profit. However, to address this threat, Proof of Stake was introduced [32]. The stake of a miner is the number of currency units that the miner possesses. In Proof of Stake, the mining capacity of a miner is restricted to the percentage of his or her stake [32]. If the miner tries to validate an invalid transaction, their share would be forfeited. Also, as all transaction information is publicly stored in the ledger, a miner cannot hide their actual stake [32].

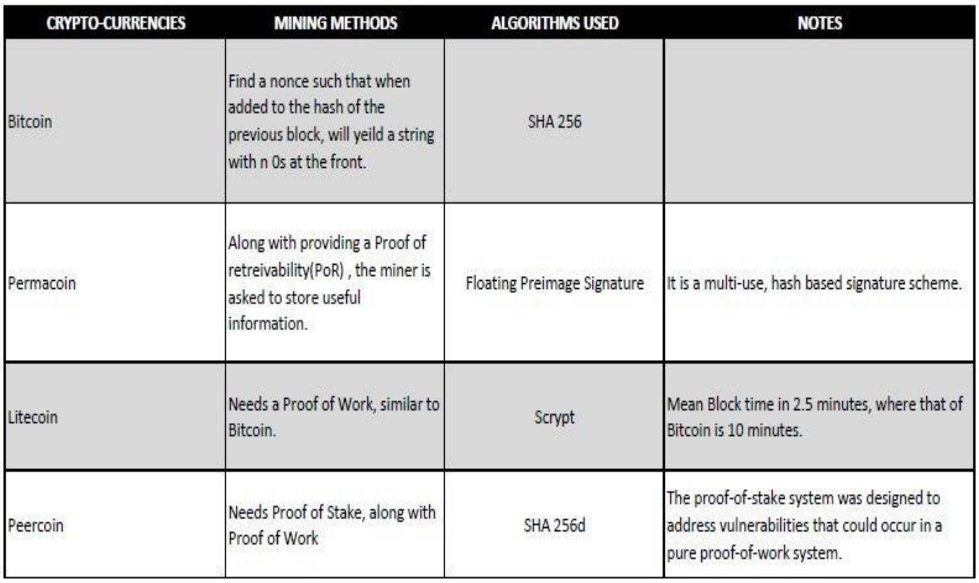
Cryptocurrencies are also vulnerable to the Sybil Attack [33] in which one user takes on multiple identities. In the Sybil Attack, attackers populate the network with fake clients controlled by them. They use them to gain a disproportionately large influence, to the point where the number of malicious nodes is greater than the number of legitimate nodes [33]. Attackers can perform the following exploits [34]:

- Disconnect legitimate nodes from the network by Denial of Service by not relaying transaction information.

- Selectively relay transaction information, exposing the victim to double spending [34].

5. Cryptocurrencies mining methods

There are many Cryptocurrency mining techniques in use. Figure 4 lists the major Cryptocurrencies and the mining algorithms they employ.



CRYPTO-CURRENCIES	MINING METHODS	ALGORITHMS USED	NOTES
Bitcoin	Find a nonce such that when added to the hash of the previous block, will yeild a string with n 0s at the front.	SHA 256	
Permacoin	Along with providing a Proof of retreivability(PoR) , the miner is asked to store useful information.	Floating Preimage Signature	It is a multi-use, hash based signature scheme.
Litecoin	Needs a Proof of Work, similar to Bitcoin.	Scrypt	Mean Block time in 2.5 minutes, where that of Bitcoin is 10 minutes.
Peercoin	Needs Proof of Stake, along with Proof of Work	SHA 256d	The proof-of-stake system was designed to address vulnerabilities that could occur in a pure proof-of-work system.

Figure 4. *Cryptocurrencies and corresponding Mining Algorithms* [35]

<u>Bitcoin</u>: Bitcoin mining uses Proof of Work [5]. The Proof of Work algorithm in use is called Hashcash[1] [36]. In Hashcash1, the miner is required to find a nonce, which, when-hashed along with the hash of the previous blocks, would yield a hash with a specified number of zeroes at its front [36]. The number of zeroes determines the difficulty metric. Mining a block is difficult because the SHA-256 hash of a block's header must be lower than or equal to the target for the block to be accepted by the network [38]. A target is a 256- bit integer shared by all Bitcoin clients, the Lower the target, the higher the difficulty. Mining is more efficient on GP-GPU than on CPUs [39]. Application Specific Integrated Circuits (ASICs) have also been developed to mine Bitcoin. Bitcoin mining works as follows [5]:

– A miner selects transactions he/she wishes to verify.
– He/she uses transactions to build a Merkle Tree[2] [41].

[1] The hash algorithm used is SHA256 [37].

[2] A Merkle tree [41] is a data structure in which every non-leaf node is labeled with the hash of the labels or values (in the case of leaves) of its child nodes. Hash trees allow efficient, secure verification of large data structures.

- Extracts root block hash from the Merkle tree.
- Adds a nonce and hashes the block header.
- Keep incrementing the nonce and hashing until the desired result is obtained [2].
- This result is the Proof of Work [27]. Other users agree/verify that the proof matches. Then the transaction is validated, and new Bitcoins are introduced.

Successful mining of coins using SHA-256 often requires hash rates at a gigahashes per second (GH/s) range or higher [39]. The current average time needed to mine a Bitcoin Block with SHA-256 is ten minutes [2].

Litecoin: Litecoin [6] was the first Cryptocurrency to use Scrypt [42] for mining. Scrypt was originally a key-derivation function (KDF) [42] developed by Percival and published in 2012. Scrypt's strength lies in the time-memory trade-off; that is, an attacker would need more memory to complete the attack faster, and Scrypt's memory requirement makes it expensive, hence slowing down any attack [6]. Scrypt has also been successfully implemented as a Proof-of-Work verification [42]; Litecoin was the first system to do so [6].

The large memory requirements of Scrypt arise from a large vector of pseudo-random bit strings generated as part of the algorithm. Once the vector is generated, its elements are accessed in a pseudo-random order and combined to produce the derived key [42]. As a Proof of Work, the key would have predefined characteristics and the miner would have to produce the sequence of bit strings that match the key [6].

Scrypt is much newer/simpler/quicker yet also more secure than the SHA-2 series [43]. While SHA is computationally intensive, Scrypt is memory intensive [43]. Scrypt's hash rates for successful coin mining generally range in the kilo hashes per second (KH/s) or mega hashes per second (MH/s) degrees of difficulty [42]. Scrypt takes only about 2.5 minutes to mine a block with the same difficulty attributes [43].

Peercoin: Peercoin [7] uses Proof of Work and introduces the concept of Proof of Stake in its mining system. Proof of Work uses the double-SHA-256 algorithm [7]. Proof of Stake also tries to reach a consensus and prevent double spending [7]. Instead of requiring the miner (known as the prover in Peercoin [7]) to perform a certain amount of computational work, a Proof of Stake system requires the prover to show ownership of a certain amount of currency [7]. Miners protect their stake in this approach [7]. With Proof of Stake, the resource compared is the amount of currency a miner holds [7] (e.g., one holding 1% of the Cryptocurrency can mine 1% of the "Proof of Stake blocks" [7]).

Proof of Stake is highly energy efficient [32]. It still has to have a block selection policy [32], inclusive of the following:

- Randomized block selection,
- Coin-age-based selection,
- Velocity-based selection, and
- Voting-based selection.

Proof of Stake, however, is said to be vulnerable to the Nothing-at-Stake Problem [32] in which miners have nothing to lose if they vote for a wrong or invalid transaction [32].

Ethereum: Ethereum was crowdfunded in 2014 [8]. Ethereum also relies on Proof of Work, but it does not use a preexisting hash algorithm [8]. The designers developed their hashing algorithm, EtHash [8, 44] (see Section VII).

The principal objective for constructing a new Proof of Work function instead of using an existing one was to mitigate the problem of mining centralization [45], in which a small group of hardware companies or mining operations can acquire a disproportionately large amount of power to impact or manipulate the network. That is ASIC-resistant [44] and has the property of memory hardness (that is, it relies on how fast the memory can move data) [44].

The Ripple Cryptocurrency uses a different approach to achieve consensus compared to traditional mining methods. Instead of using mining to validate transactions, it uses a trust-based system where each server applies the same set of transactions to the current ledger. This is done every few seconds, and the last closed ledger contains a record of all Ripple accounts and previous transactions. Transactions can be introduced by any server in the network, and servers work to reach a consensus on a set of transactions to apply to the ledger, creating a new closed ledger. [9]

Namecoin, which is known to be the first offshoot of Bitcoin, uses the same code and mining process as Bitcoin [10]. However, unlike Bitcoin, Namecoin is able to store data within its Blockchain Transaction Database [10]. In contrast, the Bitcoin Blockchain only displays the posted transactions [17] and the associated information is kept in a separate database [10].

Auroracoin: Auroracoin [11] is from Iceland. It uses Scrypt (Proof of Work) as its mining algorithm [11].

BlackCoin: BlackCoin [12] secures its network through a process called minting, which is a Proof of Stake system that validates a transaction in lesser time and is independent of Proof of Work [46].

Dash: Dash [13] (formerly Darkcoin [47]) uses a system called Darksend to add transaction privacy. Unlike other Cryptocurrencies, transaction information is not public [47]. It uses a new Proof of Work algorithm called X11 for mining- that is exclusive to Dash and is a chained hashing protocol [13]. It is claimed to be more energy efficient than Scrypt [47]. Decred: Decred [14] uses a hybrid Proof-of-Work/Proof-the of-Stake system with both miners and voters to achieve consensus. It uses Blake 256 [48] as its mining algorithm, which is a cryptographic hash function based on the ChaCha stream cipher [49].

<u>**Permacoin:**</u> Although Permacoin [15] is currently just a theoretical concept, it introduces a new approach called Proof of Retrievability [15]. This method requires miners to store a significant amount of useful information and to provide evidence to the verifier that it is being stored. Permacoin is based on large memory capacity [15], and its designers propose using storage rather than CPU cycles to secure a Cryptocurrency network [15] while also providing a practical way to backup certain data. Instead of using computational power through Proof of Work, which has no value beyond the proof itself [15].

Miller et al suggest miners store pieces of a large archive of valuable data and prove they are doing so [15]. Miners still need to solve a mathematical problem but it is less computationally intensive [15], it’s known as a scratch-off puzzle [15].

Permacoin introduces a new concept known as Proof of Retrievability [15], which requires miners to store large amounts of useful information and prove that it exists. This concept relies on large memory capacity [15] and aims to secure the network by having miners store useful data rather than consuming CPU cycles through proof of work [15]. The puzzle is based on a Floating Preimage Signature [15], which requires miners to reference a section of code stored locally on their computer to solve it [15]. If the miner successfully solves the problem, the algorithm can deduce that they are storing that data for a short period of time [15] hus.

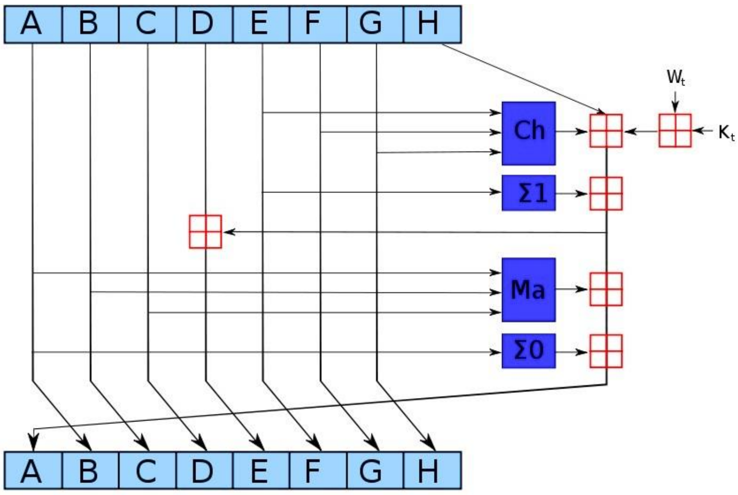


Figure 5. *The round function of SHA 256* (adopted from [18])

All miners must be storing a piece of the archived data to participate by mining Permacoin [15].

These are the major Cryptocurrencies at present, which, as we described, employ a variety of mining algorithms.

6. Hashing Algorithms

In this section, relevant hashing algorithms, namely SHA, EtHash, Scrypt and others, will be explored and their significance in the context of cryptocurrency mining will be discussed.

SHA 256: SHA 2 [50] is a set of Secure Hash Functions that has six algorithms, which produce digests (results) that are of different bit lengths. SHA256, produces a digest of 256 bits [18]. SHA 256 satisfies the requirement of unidirectional hashes (that is, any change in the input, however insignificant, leads to a completely different hash, and determining the input from the hash is practically impossible) [18]. Also, the same input will always produce the same digest [18]. SHA 256 pads input to convert its length to a multiple of 512 bits [50].

Then, it divides the input into blocks of 512 bits each [50]. The message blocks are processed one at a time, starting with a fixed initial value H0 [50], sequentially computing.

$$H^{\boldsymbol{i}} = H^{\boldsymbol{i}-1} + Ch_{Ma^{\boldsymbol{i}}}(H^{\boldsymbol{i}-1})$$

functions and Σ0 and Σ1 are bitwise rotation operators [50].

Doubled SHA 256 [37] is abbreviated as SHA256d. It is simply the SHA 256 hash performed twice serially [37]. SHA256d is used as a mining hash to increase difficulty and mining time [37]. In particular, Bitcoin uses SHA 256d [37] as its hash function, and the output is specified to have certain characteristics. For example, the N most significant bits of the digest have to zero. The miner has to come up with a nonce that, when appended to the hash of the previous block, yields a digest with this property. Other Cryptocurrencies, such as Peercoin and Namecoin, which also use SHA256d, may pose different requirements in their outputs [7].

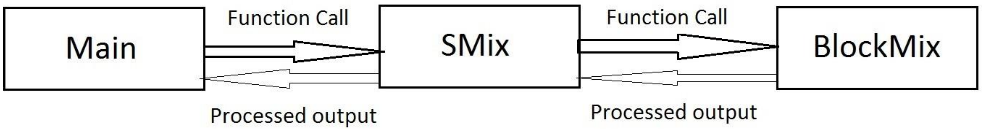


Figure 6. *Modules of Scrypt* (adopted from [42])

Scrypt: Scrypt [70], initially conceived as a Key-Derivation Function (KDF), is engineered to be resource-demanding. This characteristic is crucial in safeguarding against large-scale attacks using specialized hardware [51]. The fundamental operation of Scrypt involves processing an input to create an expansive array of pseudo-random bits. These arrays, generated in real-time, necessitate significant memory, directly impacting the speed of computation [51].

At the heart of Scrypt's architecture are two distinct functions, known as Smix and Blockmix [42]. Blockmix is tasked with executing permutation operations on the input blocks through the use of binary logical operators. Following each permutation cycle, the resultant output from Blockmix undergoes further processing by Smix, which specializes in bitwise permutations [42]. Adaptations have been made to Scrypt for its application in mining.

The use of pseudo-random bits in the original design of Scrypt results in varying outputs for the same input, posing challenges in the verification process. This level of stringent verification, however, was not a necessity in Scrypt's original role as a KDF [51]. A detailed representation of Scrypt's various modules is depicted in Figure 6.

EtHash: EtHash [44] is exclusive to Ethereum. It was designed to thwart the dominance of ASICs vis-a-vis CPUs and GPUs. The verification of the correctness of this proof of work is fast, taking .01 seconds for a light client.
The hash algorithm involves the following steps [44]:

- There exists a seed that can be computed for each block from the data stored in the block headers.
- From the seed, a 16MB pseudo-random cache can be computed. That uses its Pseudo-Random Number Generator.
- From the cache, a 1GB dataset can be generated, such that each item in the dataset depends on only a few items from the cache.
- Mining involves selecting random elements of the dataset and hashing them together. Verification can be done with low memory by using the cache to regenerate the specific pieces of the dataset that is needed, so it is sufficient to store just the cache.

Blake: Blake [52] is a cryptographic hash function based on the ChaCha stream cipher [49], but a permutation of the input block, XORed with fixed round constants, is added before each ChaCha round.

X11: X11 [53] is a chained hashing algorithm, chaining 11 different algorithms together. These are: Blake [52], BMW [54], Groestl [55], JH [56], Keccak [57], Skein [58], Luffa [59], CubeHash [60], SHAvite [61], SIMD [62], and Echo [63]. X11 is ASIC-resistant and is suitable for both CPU mining and GPU mining [53].

CryptoNight: CryptoNight [64] is a memory-intensive hash function, resistant to ASIC, GPU and FPGA architectures. CryptoNight involves three steps, generating pseudo-random addresses in a scratchpad [64], read/write operations on the addresses [64], and performing bitwise XOR and shift functions on the scratchpad [64].

SHA256 and Scrypt are the most popularly adopted mining algorithms with current Cryptocurrencies. Only a few Cryptocurrencies have developed their mining algorithms.

7. Problems encountered by Cryptocurrencies

Cryptocurrencies, while revolutionary in many aspects, have encountered a multitude of challenges and security issues throughout their relatively short history. These challenges have ranged from technical vulnerabilities to regulatory hurdles and have required the Cryptocurrency community to continuously innovate and adapt. In this section, we will explore some of the prominent problems and security challenges faced by Cryptocurrencies.

7.1. Security Breaches

One of the most high-profile security breaches in the Cryptocurrency world was the Mt. Gox hack [66]. In 2014, Mt. Gox, once the largest Bitcoin exchange in the world, declared bankruptcy after approximately 850,000 Bitcoins, worth hundreds of millions of dollars at the time, were stolen from its customers and the company itself. This incident highlighted the vulnerability of centralized exchanges and the importance of robust security measures in the Cryptocurrency ecosystem [71].

The Mt. Gox hack was not an isolated incident. Several other Cryptocurrency exchanges and wallets have fallen victim to hacking attempts, resulting in the loss of significant amounts of digital assets. These breaches underscore the need for continuous improvement in security practices within the Cryptocurrency industry [71, 72].

7.2. Smart Contract Vulnerabilities

Ethereum, a blockchain platform known for its smart contract functionality, faced a major setback with the Decentralized Autonomous Organization (DAO) incident in 2016 [45]. The DAO was a decentralized investment fund built on the Ethereum blockchain, and it became vulnerable to a flaw known as the Recursive Calling Vulnerability. This vulnerability allowed an attacker to exploit the smart contract code, leading to the siphoning of a substantial amount of ether (Ethereum's native Cryptocurrency) [73].

The DAO incident resulted in a contentious hard fork of the Ethereum blockchain to reverse the effects of the attack, leading to the creation of Ethereum (ETH) and Ethereum Classic (ETC) [73], two separate Cryptocurrencies. This event highlighted the complexities and challenges associated with governing decentralized systems and the importance of rigorous code auditing and testing for smart contracts [74].

7.3. Technical Weaknesses and Failed Projects

While Cryptocurrencies like Bitcoin and Ethereum have achieved significant success, the Cryptocurrency landscape is littered with the remnants of failed projects [65, 67, 68, 69]. Many alternative Cryptocurrencies, often referred to as "altcoins", have faced technical weaknesses and design flaws that ultimately led to their demise. Some of these issues include poor consensus mechanisms, inadequate security protocols, and low adoption rates [75].

Cryptocurrencies that have managed to succeed have had to overcome numerous obstacles, including scalability concerns, regulatory challenges, and the need to establish trust and credibility within the wider financial ecosystem.

7.4. Regulatory and Legal Challenges

The regulatory environment surrounding Cryptocurrencies remains complex and varies greatly from one jurisdiction to another. Cryptocurrencies have faced scrutiny and regulatory challenges related to money laundering, tax evasion, and fraud. Governments and regulatory bodies worldwide continue to grapple with how to regulate and classify these digital assets, which can blur the lines between currencies, commodities, and securities [76, 77].

Navigating these legal and regulatory hurdles is an ongoing challenge for Cryptocurrency projects and businesses, as they strive to strike a balance between innovation and compliance [77].

8. Future work

After finishing this general review about blockchain and hashing algorithms, my future research will delve into the innovative integration of blockchain technology and hashing algorithms within the Internet of Things (IoT). This research will focus on the application of advanced hashing algorithms as a cornerstone of blockchain technology to enhance the security and efficiency of IoT networks. I aim investigate how these algorithms can ensure data integrity and security in IoT devices and communications, addressing the pressing challenges of data tampering and unauthorized access. In addition to exploring the use of blockchain for decentralized data management in IoT ecosystems, I will also examine the scalability of hashing algorithms in diverse IoT applications, from smart homes to industrial systems. This future study will help to advance the understanding of how blockchain, fortified by robust hashing algorithms, can revolutionize IoT security, leading to more resilient and trustworthy IoT environments.

9. Conclusion

In conclusion, this paper has undertaken an in-depth exploration of the diverse mining approaches employed by major cryptocurrencies, providing insights into their unique properties and features. Presently, a dominant trend within the cryptocurrency landscape is the amalgamation of Proof of Work (PoW) and Proof of Stake (PoS) mechanisms for mining, a strategy that effectively harnesses the advantages of both approaches, the selection of Hash algorithms for PoW is a crucial determinant of mining efficiency, where memory-intensive algorithms often translate into swifter mining processes. Cryptocurrency ecosystems are dynamic and continuously seek to optimize mining procedures while also venturing into alternative consensus mechanisms. It is imperative to recognize that the cryptocurrency domain is not static but rather characterized by ongoing innovation and adaptation in response to technical advancements, economic dynamics, and user requirements. As such, it is poised to witness further evolution and transformation in the years to come, driven by a collective commitment to achieving scalability, security, and environmental sustainability.

Within this context, it remains essential for all stakeholders, including miners, developers, and users, to maintain vigilance and adaptability. The pursuit of decentralized, efficient, and secure financial systems remains at the core of the cryptocurrency movement, and it is through continued exploration and adaptation that these objectives will be realized.

In summary, cryptocurrency mining stands as a multifaceted field that reflects the dynamic nature of the digital currency ecosystem. As cryptocurrencies continue to redefine the future of finance, the evolution of mining processes and algorithms will go hand in hand, offering fresh opportunities and challenges as we advance towards a decentralized financial landscape.

References

[1] Farell, R. (2015). An analysis of the Cryptocurrency industry. https://repository.upenn.edu/handle/20.500.14332/49177.

[2] Anonymous (2014). Mining. https://en.bitcoin.it/wiki/Mining.

[3] Teutsch, J., Jain, S., Saxena, P. When Cryptocurrencies mine their own business. https://people.cs.uchicago.edu/~teutsch/papers/repurposing_miners.pdf

[4] Sklavos, N., Koufopavlou, O. (2005). Implemen- tation of the sha-2 hash family standard using fpgas. *The Journal of Supercomputing*, 31 (3), pp. 227–248. https://doi.org/ 10.1007/s11227-005-0086-5

[5] Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system. https://bitcoin.org/bitcoin.pdf.

[6] Lacity, M. C. (2022). Blockchain: from bitcoin to the internet of value and beyond. *Journal of Information Technology*, 37(4), pp. 326-340.

[7] King, S., Nadal, S. (2012). Ppcoin: Peer-to-peer crypto-currency with proof-of-stake. self-published pa-per, August, 19, 2012.

[8] Wood, G. (2014). *Ethereum: A secure decentralised gener- alised transaction ledger*. Ethereum Project Yellow Paper.

[9] Schwartz, D., Youngs, N., Britto, A. (2014). *The ripple protocol consensus algorithm*. Ripple Labs Inc White Paper, p. 5. https://api.semanticscholar.org/CorpusID:26971000.

[10] Kalodner, H., Carlsten, M., Ellenbogen, P., Bonneau, J., Narayanan, A. (2015). *An empirical study of namecoin and lessons for decentralized namespace design*. Technical report, Citeseer.

[11] Cawrey, D. (2014). Auroracoin airdrop: Will iceland embrace a national digital currency. *CoinDesk*, March, 24, 2014.

[12] Vasin, P. (2014). Blackcoin's proof-of-stake protocol v2. https://www.dailyblackcoin.com/blackcoin-pos-protocol-v2-whitepaper.pdf

[13] anonymous (2014). *Dash*. https://www.dash.org/wp-content/uploads/2015/04/Dash - WhitepaperV1.pdf.

[14] anonymous (2014). *Decred*. https://decred.org/.

[15] Miller, A., Juels, A., Shi, E., Parno, B., Katz, J. (2014). Permacoin: Repurposing bitcoin work for data preservation. In: *Security and Privacy (SP), 2014 IEEE Symposium on*, IEEE, pp. 475–490. https://doi.org/ 10.1109/SP.2014.37

[16] Swan, M. (2015). Blockchain: *Blueprint for a new economy*. O'Reilly Media, Inc. https://doi.org/10.5555/3006358

[17] Anonymous (2014). Blockchain. http://www.investopedia.com/ terms/b/Blockchain.asp.

[18] Rogaway, P., Shrimpton, T. (2004). Cryptographic hash-function basics: Definitions, implications, and separations for preimage resistance, second-preimage resistance, and collision resistance. In: *International Workshop on Fast Software Encryption*, Springer, pp. 371–388. https://doi.org/10.1007/978-3-540-25937-4_24

[19] Ahamad, S-S., Nair, M., Varghese, B. (2013). A survey on crypto currencies. In: *4th International Conference on Advances in Computer Science, AETACS*, Citeseer, pp. 42–48. https://doi.org/10.21474/IJAR01/13608

[20] Chaum, D. (1997). David chaum on electronic commerce how much do you trust big brother? *IEEE Internet Computing*, 1 (6), pp. 8–16. https://doi.org/10.1109/MIC.1997.643931

[21] Eisenmann, T. R., Barley, L. (2006). Paypal mer- chant services. Available at hbs.edu.

[22] Jack, W., Suri, T. (2011). *Mobile money: The economics of m-pesa*. Technical report, National Bureau of Economic Research. https://ideas.repec.org/p/nbr/nberwo/16721.html.

[23] Mankiw, N. G. (2014). Principles of macroeconomics. Cengage Learning.

[24] Loera, A. (2014). *Method of making, securing, and using a Cryptocurrency wallet*. February 11, 2014, US Patent App. 14/178,234.

[25] Anonymous (2014). *Block*. https://en.bitcoin.it/wiki/Block.

[26] *Double spending*. https://en.bitcoin.it/wiki/ Double-spending.

[27] Anonymous (2014). *Proof of work*. https://en.bitcoin.it/wiki/ Proof of work.

[28] Anonymous (2014). *Limit*. http://bitcoin. stackexchange.com/questions/161/how-many-bitcoins-will-there-eventually-be# comment7700 274.

[29] Bentov, I., Lee, C., Mizrahi, A., Rosenfeld, M. (2014). Proof of activity: Extending bitcoin's proof of work via proof of stake [extended abstract]. *ACM SMETRICS Performance Evaluation Review*, 42 (3), pp. 34–37. https://eprint.iacr.org/2014/452

[30] Bershidsky, L. (2014). Did Ukrainians Almost Take Over Bitcoin. BloombergView, http://www. bloombergview. com/articles/2014-01-14/did-ukrainians-almost-take-over-bitcoin.

[31] Anonymous (2014). *Fifty one percent*. https://en.bitcoin.it/wiki/ Weaknesses#Attacker has a lot of computing power.

[32] Anonymous (2014). *Proof of stake*. https://en.bitcoin.it/wiki/ Proof of Stake.

[33] Douceur, J. R. (2002). The sybil attack. In: *Peer-to-peer Systems*, Springer, pp. 251–260. https://doi.org/10.1007/3-540-45748-8_24

[34] Levine, B. N., Shields, C., Margolin, N. B. (2006). *A survey of solutions to the sybil attack*. University of Massachusetts Amherst, Amherst, MA, 7.

[35] Anonymous (2014). *Market capitalization of Cryptocurrencies*. https://coinmarketcap.com/.

[36] Raikwar, M., Gligoroski, D., & Kralevska, K. (2019). SoK of used cryptography in blockchain. *Ieee Access*, 7, 148550-148575. https://doi.org/10.1109/ACCESS.2019.2946983

[37] Courtois, N. T., Grajek, M., Naik, R. (2014). Optimizing sha256 in bitcoin mining. In: *Cryptography and Security Systems*, Springer, pp. 131–144. https://doi.org/10.1007/978-3-662-44893-9_12

[38] Anonymous (2014). *Mining difficulty metric*. https://en.bitcoin.it/wiki/Mining#The Difficulty Metric.

[39] O'Dwyer, K. J., Malone, D. (2013). Bitcoin mining and its energy footprint. In: *Irish Signals & Systems Conference 2014 and 2014 China-Ireland International Conference on Information and Communications Technologies (ISSC 2014/CIICT 2014). 25th IET, IET, 2013*, pp. 280–285. https://doi.org/10.1049/cp.2014.0699

[40] Polkinghorne, J., Desnoyers, M. (1989). *Application specific integrated circuit*. March 28, 1989. US Patent 4,816,823.

[41] Okupski, K. (2014). *Bitcoin developer reference*. http://enetium.com/resources/ Bitcoin. pdf.

[42] Percival, C., Josefsson, S. (2015). *The scrypt password-based key derivation function*. https://api.semanticscholar.org/CorpusID:31567403

[43] Anonymous (2014). *Sha2 and scrypt*. https://www.coinpursuit.com/pages/bitcoin-altcoin-SHA-256-scrypt-mining-algorithms/, 2014.

[44] Anonymous (2014). *Ethash*. https://github.com/ethereum/wiki/ wiki/Ethash.

[45] Buterin, V. (2014). *A next-generation smart contract and decentralized application platform*. white paper.

[46] Croteau, M., Litranab, E. (2014). *Proof of stake: Def-inite. an implementation of constant staking rewards to promote increased network activity*. https://doi.org/10.1109/PST.2016.7906988

[47] Duffield, E., Hagan Darkcoin K. (2014). *Peertopeer Cryptocurrency with anonymous blockchain transactions and an improved proof-of-work system*. https://api.semanticscholar.org/CorpusID:2236856.

[48] Aumasson, J.-P., Neves, S., Wilcox-O'Hearn, Z., Winnerlein, C. (2013). Blake2: simpler, smaller, fast as md5. In: *International Conference on Applied Cryptography and Network Security*, Springer. https://doi.org/10.1007/978-3-642-38980-1_8

[49] Bernstein, D. J. (2008). Chacha, a variant of salsa20. In: *Workshop Record of SASC*, Vol. 8. https://eprint.iacr.org/2007/472.pdf

[50] Gilbert, H., Handschuh, H. (2003). Security analysis of sha-256 and sisters. In: *Selected areas in cryptography*. Springer, pp. 175–193. https://doi.org/10.1007/978-3-540-24654-1_13

[51] Percival, C. (2009). *Stronger key derivation via sequential memory-hard functions*. Self-published, pp. 1–16. https://api.semanticscholar.org/CorpusID:15333875

[52] Dunkelman, O., Khovratovich, D. (2011). Iterative differentials, symmetries, and message modification in blake-256. In: *ECRYPT2 Hash Workshop*, Vol. 2011. Citeseer. https://eprint.iacr.org/2016/827.pdf

[53] Anonymous (2014). *Hashx11*. http://cryptorials.io/glossary/x11/.

[54] El-Hadedy, M., Margala, M., Gligoroski, D., Knapskog, S. J. (2010). Resource-efficient im- plementation of blue midnight wish-256 hash function on xilinx fpga platform. In: *Information Assurance and Security (IAS), 2010 Sixth International Conference on*, IEEE, pp. 44–47. https://doi.org/10.1109/ISIAS.2010.5604066

[55] Jungk, B., Reith, S., Apfelbeck, J. (2009). On optimized fpga implementations of the sha-3 candidate groestl. *IACR Cryptology ePrint Archive*, 2009, 206, https://eprint.iacr.org/2009/206.

[56] Wu, H. (2011). The hash function jh. *Submission to NIST* (Round 3), p. 6.

[57] Bertoni, G., Daemen, J., Peeters, M., Van Assche, G. (2011). The keccak sha-3 submission. *Submission to NIST* (Round 3), 6 (7), p. 16.

[58] Ferguson, N., Lucks, S., Schneier, B., Whiting, D., Bellare, M., Kohno, T., Callas, J., Walker, J. (2010). The skein has a function family. *Submission to NIST* (Round 3), 7 (7, 5), p. 3. https://api.semanticscholar.org/CorpusID:59739596

[59] De Canniere, C., Sato, H., Watanabe, D. (2009). Hash function luffa: specification. *Submission to NIST* (Round 2). http://ehash.iaik.tugraz.at/uploads/e/ea/Luffa_Specification.pdf.

[60] Bernstein, D. J. (2008). Cubehash specification (2. b. 1). *Submission to NIST*. https://doi.org/10.1007/978-3-642-21554-4_27

[61] Biham, E., Dunkelman, O. (2009). The shavite-3 hash function. *Submission to NIST* (Round 2), p. 113. https://api.semanticscholar.org/CorpusID:6386607

[62] Tillich, S., Feldhofer, M., Kirschbaum, M., Plos, T., Schmidt, J.-M., Szekely, A. (2009). High-speed hardware implementations of blake, blue midnight wish,

cubehash, echo, fugue, gro¨stl, hamsi, jh, keccak, luffa, shabal, shavite-3, simd, and skein. *IACR Cryptology ePrint Archive*, 2009, p. 510. https://eprint.iacr.org/2009/510

[63] Schläffer, M. (2010). Subspace distinguisher for 5/8 rounds of the echo-256 hash function. In: *International Workshop on Selected Areas in Cryptography*, Springer, pp. 369–387. https://doi.org/10.1007/978-3-642-19574-7_25

[64] Anonymous (2014). *Cryptonight*. https://cryptonote.org/cns/ cns008.txt.

[65] Luther, W. J. (2015). Cryptocurrencies, network effects, and switching costs. *Contemporary Economic Policy*. https://doi.org/10.1111/coep.12151

[66] Karpeles, M. (2011). *Clarification of mt. gox compromised accounts and major bitcoin sell-off*. https://bitcointalk.org/index.php?topic=24727.0

[67] Anonymous (2014). Failed Cryptocurrencies. https://ltcflux.wordpress.com/2013/04/11/failed-Cryptocurrencies-do-exist/.

[68] Lee, D. K. C. (2014). *The Cryptocurrency revolution and its impact*. Self-published. https://ink.library.smu.edu.sg/podcasts/28/

[69] Anonymous (2014). *Fail qubic*. https://bitcointalk.org/index.php? topic=112676.0.

[70] Tomar, A., Agarwal, P.: A Comprehensive Study of Cryptocurrency Systemshttps://www.academia.edu/37069857/A_Comprehensive_Study_of_Cryptocurrency_Systems

[71] Rao, S. (2021). Mt. Gox – The fall of a giant. Understanding Cryptocurrency Fraud, pp. 71–82. https://doi.org/10.1515/9783110718485-006

[72] Williams, M., Çekin, S. E., & Green, D. (2020).Bitcoin and the Cross-Market Effects of the Mt. Gox Meltdown." Issues In Information Systems, 2020. Crossref. https://doi.org/10.48009/3_iis_2020_245-252.

[73] Mehar, Muhammad Izhar et al. (2019). Understanding a revolutionary and flawed grand experiment in blockchain: the DAO attack. *Journal of Cases on Information Technology (JCIT)*, 21, 1, pp. 19–32. https://doi.org/10.4018/JCIT.2019010102

[74] Dhillon, V., Metcalf, D., Hooper, M. (2021). The DAO Hacked. In: *Blockchain Enabled Applications*. Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4842-6534-5_6

[75] Yi, Xiao et al. (2021). Diving into Blockchain's Weaknesses: An Empirical Study of Blockchain System Vulnerabilities arXiv preprint arXiv:2110.12162. https://doi.org/10.48550/arXiv.2110.12162

[76] Yeoh, P. (2017). Regulatory issues in blockchain technology. *Journal of Financial Regulation and Compliance*, Vol. 25, No. 2, pp. 196–208. https://doi.org/10.1108/JFRC-08-2016-0068

[77] Fulmer, Nathan (2018). Exploring the legal issues of blockchain applications. *Akron L. Rev.*, 52, p. 161. https://ideaexchange.uakron.edu/akronlawreview/vol52/iss1/5.