A Brief Survey of Cryptocurrency Systems

Ujan Mukhopadhyay*, Anthony Skjellum*, Oluwakemi Hambolu†, Jon Oakley†, Lu Yu† and Richard Brooks†
*Auburn Cyber Research Center
Auburn University
Email: {uzm0002,skjellum}@auburn.edu
†Dept. of Electrical and Computer Engineering
Clemson University
{ohambol,joakley,ylu,rrb}@g.clemson.edu

***Abstract*—Cryptocurrencies have emerged as important financial software systems. They rely on a secure distributed ledger data structure; mining is an integral part of such systems. Mining adds records of past transactions to the distributed ledger known as Blockchain, allowing users to reach secure, robust consensus for each transaction. Mining also introduces wealth in the form of new units of currency. Cryptocurrencies lack a central authority to mediate transactions because they were designed as peer-to-peer systems. They rely on miners to validate transactions. Cryptocurrencies require strong, secure mining algorithms.**

In this paper we survey and compare and contrast current mining techniques as used by major Cryptocurrencies. We evaluate the strengths, weaknesses, and possible threats to each mining strategy. Overall, a perspective on how Cryptocurrencies mine, where they have comparable performance and assurance, and where they have unique threats and strengths are outlined.

***Index Terms*—Cryptocurrency, Mining, Blockchain**

I. Introduction

A Cryptocurrency is a peer-to-peer digital exchange system in which cryptography is used to generate and distribute currency units [1]. This process requires distributed verification of transactions without a central authority. Transaction verification confirms transaction amounts, and whether the payer owns the currency they are trying to spend while ensuring that currency units are not spent twice. This verification process is called *mining* [2]. Cryptocurrencies use a variety of mining technologies, according to their particular requirements. For instance, certain Cryptocurrencies focus on restricting the number of transactions validated per unit time, while others concentrate on achieving fast, lightweight services [3]. Some mining algorithms are deliberately memory intensive; others are computationally expensive [4]. In this paper, we survey Cryptocurrency mining systems and analyze their efficiency.

Cryptocurrency systems considered in this paper are as follows: 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 are the most interesting, widely used, and with the greatest capital and transaction rates. Also, they showcase the major mining algorithms.

The remainder of this paper is organized as follows: Section II defines relevant terms. Section III provides historical perspective and background. Section IV provides an overview of Blockchains [16]. Section V overviews mining, while Section VI provides further details. Section VII discusses relevant Hash algorithms. Section VIII addresses problems encountered with Cryptocurrencies. Section IX offers conclusions.

II. Terminology

Key terms used in this paper include the following:

- **Block:** a data structure containing transaction data.
- **Blockchain**: a public ledger of all transactions that have ever been executed [17]. It consists of a distributed, chronological chain of blocks and constantly growing as *completed* blocks are added to it with a new set of records. Blocks comprise transactions and information from previous blocks. A unique linear path from the first block ever posted to the current block exists because every block includes the hash of the previous block.
- **Mining**: a required verification step for a Cryptocurrency transaction and for adding transaction records to the public ledger (the Blockchain). Mining also introduces new Cryptocurrency units in the system [2].
- **Hash**: a one-way function [18] that takes data of any size as input and produces a fixed length output. Hash computation should be fast and easy, while reversing the process should be expensive and difficult. Reversal should require a brute force algorithm. Any change in the input should propagate through the entire output, so that outputs for similar input have no predictable similarity.
- **Nonce**: a number, usually chosen at random used once for a specific purpose then discarded. Nonce collisions, which happen when two randomly chosen nonces turn out to be the same, are ignored.
- **Fork**: a quantity generated when two blocks are created a few seconds apart [17]. Forks are resolved by adding the block received first to the Blockchain. Subsequent Blocks are added to the included Block.

III. 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 key 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

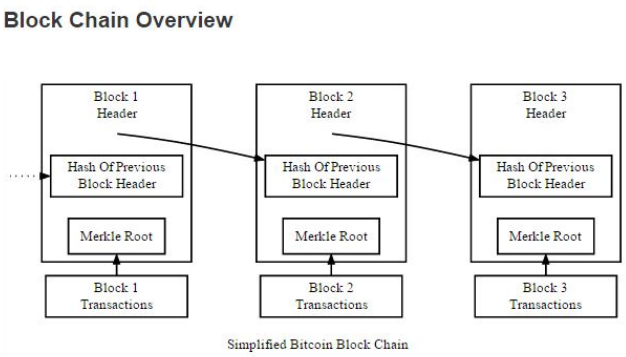

Figure 1. A Bitcoin Blockchain (adopted from [5])

be linked with bank accounts and credit cards, and users can pay someone or receive payment through the PayPal accounts. PayPal does not have a its own currency.

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 the user can deposit money into an account stored in their cell phones and send PIN-secured SMS texts to other users in order 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 using the payer's private key.
- The transaction is verified by mining [2].

IV. Blockchain Overview

A Blockchain is a distributed public ledger of Cryptocurrency transactions [17]. Each verified transaction is accumulated in a block [25]. Each block consists of a variable number of verified transactions. The maximum size of a block is fixed in each Cryptocurrency system, providing an upper bound to the number of transactions included. For instance, the maximum size of a Bitcoin [5] block is 1MB. Figure 1 shows a simplified representation of a Bitcoin Blockchain.

A Bitcoin Block consists 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.

The first block ("genesis block") contains the first transactions of a given Cryptocurrency. The hash of the first block

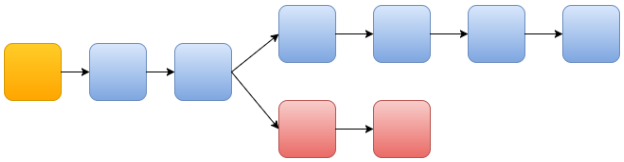

Figure 2. Forking in a Blockchain (adopted from [17])

is passed forward to the miner, which uses it and generates a nonce to create a hash for the second block. Likewise, the hash of the third block contains the hashes of the first two blocks, and so on. Thus, a strict chronological link, or chain, is created from the genesis block to the current block through the inclusion of hashes. There is a single, unique path from the most recent block to that first block. This relationship makes it extremely difficult for an attacker to tamper with the information in a block, because all subsequent blocks would have to be regenerated, which would be detected because the final hash wouldn't 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.

V. Mining Overview

Every Cryptocurrency system that we've 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 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 the Blockchain. A malicious user may create multiple nodes and try to validate an invalid transaction. To prevent this, miners are required to solve a resource-intensive task. Resource intensiveness 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 an easily verifiable result of a resource intensive task that confirms that the task has been performed.
- *Proof of Stake* [7], which requires the miner to show how much currency the miner owns in the system.
- *Proof of Retrievability* [15], which requires the miner to show that the data he was given to store is intact and can be recovered 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 requirement also restricts the number of transactions that can be validated (and consequently the number of blocks added to the ledger) in a given time period. This restriction is necessary because, with each block mined, new currency units—the total of which is finite—are

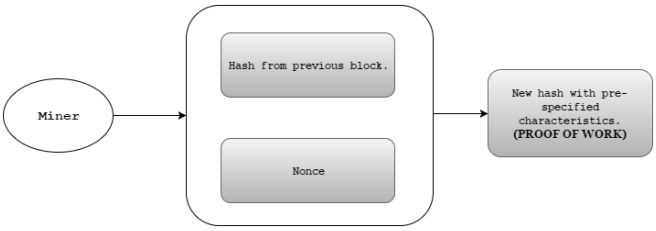

Figure 3. A simplified view of Proof of Work mining (adapted from [2])

produced. Thus, it is necessary to slow down the rate of the production to prevent untimely exhaustion. 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 were not 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 remains as a potential future event for Bitcoin).

Here are the steps involved in mining:

- A miner performs a resource-intensive task and produces a proof that the work has been done [27]. This task prevents a malicious miner from forming false identities and manipulating. Figure 3 offers a simplified view.
- The proof produced is verified to confirm that the task has been performed.
- The miner then checks for 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 in order 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]. As input, the miner gets the hash of the previous blocks. He/she would have to 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 in a way so that determining the input from the output is extremely time consuming so as to be intractable [18]. The miner has to 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. Obtaining the correct nonce is resource-intensive as well as time consuming since it involves calculating a huge number of hashes, whereas verifying if indeed 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 detected in the proof of work system was the *51% attack* [30]. If a single entity occupies 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 amount of currency unit 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 spurious clients controlled by them. They use them to gain a disproportionately large influence to the point where the number of malicious nodes are 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].

VI. Cryptocurrency Mining Methods

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

Bitcoin: Bitcoin mining uses Proof of Work [5]. The Proof of Work algorithm in use is called Hashcash [36]. In Hashcash[1], the miner is required to find a nonce, which, when

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

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.
Ripple	Does not use mining. Uses a trust based consensus system.	Elliptic Curve Digital Signature Algorithm	A transaction is any proposed change to the ledger and can be introduced by any server to the network. The servers attempt to come to consensus about a set of transactions to apply to the ledger, and the goal of consensus is for each server to apply the same set of transactions to the current ledger.
Ethereum	Proof of Work	EtHash	EtHash was developed by the Ethereum project.
Blackcoin	Minting	Scrypt	Blackcoin uses scrypt as well as Proof of Stake in a process called minting, and does not use Proof of Work.
Dash	Proof of Work	X11	Previously known as Darkcoin, Dash instills more privacy to the transaction by Darksend. X11 is exclusive to Dash and is more energy efficient than Scrypt.
Auroracoin	Proof of Work	Scrypt	Developed and used in Iceland as an alternative currrency.
Decred	Proof of Work and Proof of Stake	Blake-256	
Primecoin	Proof of Work	Cunningham Chain	Uses Cunningham chain and Bi-Twin Chain for Proof of Work.

Figure 4. Cryptocurrencies and corresponding Mining Algorithms [35]

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 determine 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 in order 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, higher the difficulty. Mining is more efficient on GP-GPU than in CPUs [39]. Application Specific Integrated Circuits (ASICs) [40] 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].
- Extracts root block hash from the Merkle tree.
- Adds a nonce, hashes the block header.
- Keeps 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].

[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 case of leaves) of its child nodes. Hash trees allow efficient, secure verification of large data structures.

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 kilohashes per second (KH/s) or megahashes 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. For Proof of Work, its 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 own 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 own 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. EtHash is ASIC-resistant [44], and has the property of memory hardness (that is, it relies on how fast the memory can move data) [44].

Ripple: Ripple [9] does not use mining in its truest sense. Released in 2012, it uses a trust-based system to attain consensus [9]. The goal of consensus is for each server to apply the same set of transactions to the current ledger [9]. A new ledger is created every few seconds and the last closed ledger contains a perfect record of all Ripple accounts and previous transactions [9]. A transaction is any proposed change to the ledger; it can be introduced by any server in the network [9]. Servers try to reach consensus about a set of transactions to apply to the ledger, creating a new last closed ledger [9].

Namecoin: Namecoin [10] is known to be the first branch of Bitcoin. It is a branch in the sense that Namecoin utilizes the same code and mining algorithm as Bitcoin [10]. Unlike Bitcoin, Namecoin can store data in its own Blockchain Transaction Database [10]. The Bitcoin Blockchain shows the posted transactions only [17]; the related information is stored 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 [47] 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-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].

Permacoin: While Permacoin [15] is theoretical, without any known implementation at present, it introduces a new concept of Proof of Retrievability [15]. This scheme requires that the miner store some useful information (of considerable size) and present a proof to the verifier that it exists.

Permacoin relies on large memory capacity [15]. The designers proposes using storage, rather than CPU cycles, to secure a Cryptocurrency network [15] while providing a useful way to back up certain data in the process. Instead of consuming cycles through Proof of Work, which has no intrinsic value beyond the proof itself [15], Miller et al want miners to store pieces of a large archive of data that is worth preserving [15]. They suggest this can be achieved by having miners prove that they are storing those pieces of data [15]. Miners still have to prove that they have solved a mathematical problem but it is much less computationally intensive [15], and is known as a scratch-off puzzle [15].

The puzzle is based on a Floating Preimage Signature [15]. The miners must refer to a section of code stored locally on their computer to solve the puzzle [15]. If they successfully solve the problem, then the algorithm can deduce that they are storing that data (at least for a short time) [15]. Thus, all miners must be storing a piece of the archived data to participate by mining Permacoin [15].

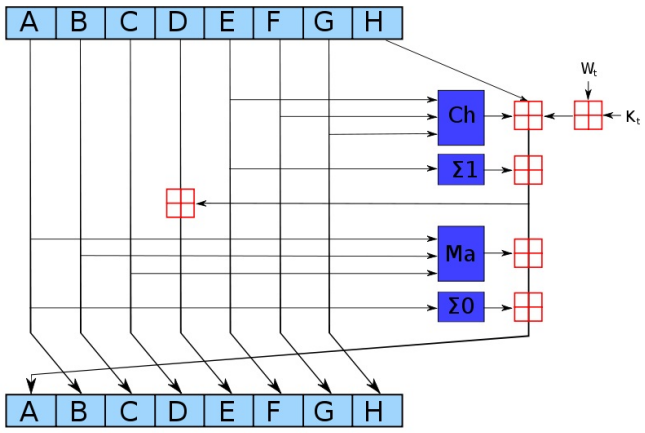

Figure 5. Round function of SHA 256 (adopted from [18])

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

VII. Hash Algorithms

In this section, relevant Hash algorithms, namely:

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. SHA 256, 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 H^0 [50] , sequentially computing

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

where Ch is the SHA-256 compression function and $+$ means word-wise addition modulo 2^{32} [50]. H^N becomes the hash [50]. The compression function permutes and compresses the input block ans is a combination of bitwise logical operators, such as AND, OR, XOR, Complement, etc [50]. In Figure 5, Ch and Ma are the blockwise logical operators using XOR functions, and $\Sigma 0$ and $\Sigma 1$ 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 a certain characteristics. For example, the N most significant bits of the digest have to be zeroes. 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

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

as Peercoin and Namecoin, that also use SHA256d, may pose different requirements in their outputs [7].

Scrypt: Scrypt [42] was designed to be a Key-Derivation Function (KDF). All Key-Derivation Functions are resource intensive in order to mitigate large-scale custom hardware attacks [51]. Scrypt takes an input and *generates* a large vector of pseudo-random bits. Since these vectors are generated at runtime, the algorithms require large memory. More memory leads to faster computation [51].

Within the algorithm, there are two functions called Smix and Blockmix [42]. Blockmix performs permutation operations on the input blocks using binary logic operands and, in each iteration, the output of the Blockmix is again processed in Smix, which performs bitwise permutations [42]. Scrypt was modified for the purposes of mining. Since the original Scrypt uses pseudo-random bits, the outputs of the same input would be different. This makes it harder to verify. When Scrypt was used for KDF, there was no need for such verification [51]. Figure 6 shows the different modules of Scrypt.

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 correctness of this proof of work is fast, taking .01 seconds for a light client.

The EtHash 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. EtHash uses its own 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 own mining algorithms.

VIII. Problems Encountered By Cryptocurrencies

Existing Cryptocurrencies have faced various problems and security issues [65] thus far, including these:

- There was a Bitcoin Exchange in Tokyo, known as Mt. Gox [66]. It was developed in 2010 to help users exchange Bitcoins with regular currency. But Mt. Gox had a security breach resulting in a momentary drop of the Bitcoin price to one cent [66]. Although the price was soon restored and stabilized, many Bitcoins were lost. In 2011, they mistakenly sent over 2,500 Bitcoins to invalid addresses thereby exposing faults in their protocol [66]. Although Mt. Gox handled 70% of bitcoin transactions in 2013 [66], they declared bankruptcy after 850,000 Bitcoins were stolen from their customers and the company itself [66] in a second hack.
- Ethereum has a Decentralized Autonomous Organization (DAO) [45] in the Ethereum Blockchain that facilitates validation. Recently, the DAO was exposed to the Recursive Calling Vulnerability, in which an attacker called a function to split a transaction recursively and collected ether, the Ethereum currency [45].
- There have been certain alternative Cryptocurrencies that have failed [67]. Some of these weren't sustained or elaborate efforts, including BBQCoin [68] and Solidcoin [68]. Some others failed to take precautions and succumbed either to the 51% and/or Sybil Attacks [65].
- Qubic was developed as an inspiration from Bitcoin, but without the disadvantages of Bitcoin. They proposed a network-based Proof of Work instead of a CPU-based one. However, Qubic wasn't popular and was shut down [69].

Technical weaknesses and flawsresulted in the failure of several Cryptocurrencies. The Cryptocurrencies that have prevailed thus far have had to overcome snags too.

IX. Conclusion

This paper compared, contrasted, and surveyed major Cryptocurrencies' approaches to mining, as well as other properties and features of the systems. At present, major Cryptocurrencies use Proof of Work, Proof of Stake or a combination

thereof for mining. While Proof of Work is resource intensive, Proof of Stake cannot act independently. A combination of the both is found to be effective. For Proof of Work, Cryptocurrencies use various Hash algorithms. A majority of these are CPU-intensive and the others are memory intensive. Typically memory-intensive hash functions have been found to be faster mining algorithms.

Cryptocurrencies are experimenting with their mining protocols and algorithms to optimize their performance, and some are trying to identify alternatives to mining.

X. Acknowledgement

This material is based upon work supported by the National Science Foundation under Grants Nos. 1547164 and 1547245. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

A thorough review of a late draft of this paper by Dr. Purushotham V. Bangalore of UAB is kindly acknowledged.

References

[1] Ryan Farell. An analysis of the cryptocurrency industry. *available at repository.upenn.edu*, 2015.

[2] anonymous. Mining. https://en.bitcoin.it/wiki/Mining, 2014.

[3] Jason Teutsch, Sanjay Jain, and Prateek Saxena. When cryptocurrencies mine their own business.

[4] Nicolas Sklavos and Odysseas Koufopavlou. Implementation of the sha-2 hash family standard using fpgas. *The Journal of Supercomputing*, 31(3):227–248, 2005.

[5] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system, 2008.

[6] C Lee. Litecoin, 2011.

[7] Sunny King and Scott Nadal. Ppcoin: Peer-to-peer crypto-currency with proof-of-stake. *self-published paper, August*, 19, 2012.

[8] Gavin Wood. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum Project Yellow Paper*, 2014.

[9] David Schwartz, Noah Youngs, and Arthur Britto. The ripple protocol consensus algorithm. *Ripple Labs Inc White Paper*, page 5, 2014.

[10] Harry Kalodner, Miles Carlsten, Paul Ellenbogen, Joseph Bonneau, and Arvind Narayanan. An empirical study of namecoin and lessons for decentralized namespace design. Technical report, Citeseer, 2015.

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

[12] Pavel Vasin. Blackcoin's proof-of-stake protocol v2, 2014.

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

[14] anonymous. Decred. https://decred.org/, 2014.

[15] Andrew Miller, Ari Juels, Elaine Shi, Bryan Parno, and Jonathan Katz. Permacoin: Repurposing bitcoin work for data preservation. In *Security and Privacy (SP), 2014 IEEE Symposium on*, pages 475–490. IEEE, 2014.

[16] Melanie Swan. *Blockchain: Blueprint for a new economy*. " O'Reilly Media, Inc.", 2015.

[17] anonymous. Blockchain. http://www.investopedia.com/terms/b/Blockchain.asp, 2014.

[18] Phillip Rogaway and Thomas Shrimpton. Cryptographic hash-function basics: Definitions, implications, and separations for preimage resistance, second-preimage resistance, and collision resistance. In *International Workshop on Fast Software Encryption*, pages 371–388. Springer, 2004.

[19] ShaikShakeel Ahamad, Madhusoodhnan Nair, and Biju Varghese. A survey on crypto currencies. In *4th International Conference on Advances in Computer Science, AETACS*, pages 42–48. Citeseer, 2013.

[20] David Chaum. David chaum on electronic commerce how much do you trust big brother? *IEEE Internet Computing*, 1(6):8–16, 1997.

[21] Thomas R Eisenmann and Lauren Barley. Paypal merchant services. *Available at hbs.edu*, 2006.

[22] William Jack and Tavneet Suri. Mobile money: The economics of m-pesa. Technical report, National Bureau of Economic Research, 2011.

[23] N Gregory Mankiw. *Principles of macroeconomics*. Cengage Learning, 2014.

[24] Anthony Loera. Method of making, securing, and using a cryptocurrency wallet, February 11 2014. US Patent App. 14/178,234.

[25] anonymous. Block. https://en.bitcoin.it/wiki/Block, 2014.

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

[27] anonymous. Proof of work. https://en.bitcoin.it/wiki/Proof_of_work, 2014.

[28] anonymous. Limit. http://bitcoin.stackexchange.com/questions/161/how-many-bitcoins-will-there-eventually-be#comment7700_274, 2014.

[29] Iddo Bentov, Charles Lee, Alex Mizrahi, and Meni Rosenfeld. Proof of activity: Extending bitcoin's proof of work via proof of stake [extended abstract] y. *ACM SIGMETRICS Performance Evaluation Review*, 42(3):34–37, 2014.

[30] Gavin Andresen. Neutralizing a 51% attack. *Electrónica. Back, A.(2002, Agosto). Hashcash-a denial of service counter-measure. Electrónica. Bershidsky, L.(2014, Enero). Did ukrainians almost take over bitcoin*, 2012.

[31] anonymous. Fifty one percent. https://en.bitcoin.it/wiki/Weaknesses#Attacker_has_a_lot_of_computing_power, 2014.

[32] anonymous. Proof of stake. https://en.bitcoin.it/wiki/Proof_of_Stake, 2014.

[33] John R Douceur. The sybil attack. In *Peer-to-peer Systems*, pages 251–260. Springer, 2002.

[34] Brian Neil Levine, Clay Shields, and N Boris Margolin.

A survey of solutions to the sybil attack. *University of Massachusetts Amherst, Amherst, MA*, 7, 2006.

[35] anonymous. Market capitalization of cryptocurrencies. https://coinmarketcap.com/, 2014.

[36] Adam Back. The hashcash proof-of-work function. *Draft-Hashcash-back-00, Internet-Draft Created,(Jun. 2003)*, 2003.

[37] Nicolas T Courtois, Marek Grajek, and Rahul Naik. Optimizing sha256 in bitcoin mining. In *Cryptography and Security Systems*, pages 131–144. Springer, 2014.

[38] anonymous. Mining difficulty metric. https://en.bitcoin.it/wiki/Mining#The_Difficulty_Metric, 2014.

[39] Karl J O'Dwyer and David Malone. 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*, pages 280–285. IET, 2013.

[40] John Polkinghorne and Michael Desnoyers. Application specific integrated circuit, March 28 1989. US Patent 4,816,823.

[41] Krzysztof Okupski. Bitcoin developer reference. *Available at http://enetium. com/resources/Bitcoin. pdf*, 2014.

[42] Colin Percival and Simon Josefsson. The scrypt password-based key derivation function. *available at ietf.org*, 2015.

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

[44] anonymous. Ethash. https://github.com/ethereum/wiki/wiki/Ethash, 2014.

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

[46] Mike Croteau and Emir Litranab. Proof of stake: Definite. an implementation of constant staking rewards to promote increased network activity, 2014.

[47] Evan Duffield and Kyle Hagan. Darkcoin: Peertopeer cryptocurrency with anonymous blockchain transactions and an improved proofofwork system. *available at bitpaper.info*, 2014.

[48] Jean-Philippe Aumasson, Samuel Neves, Zooko Wilcox-O'Hearn, and Christian Winnerlein. Blake2: simpler, smaller, fast as md5. In *International Conference on Applied Cryptography and Network Security*, pages 119–135. Springer, 2013.

[49] Daniel J Bernstein. Chacha, a variant of salsa20. In *Workshop Record of SASC*, volume 8, 2008.

[50] Henri Gilbert and Helena Handschuh. Security analysis of sha-256 and sisters. In *Selected areas in cryptography*, pages 175–193. Springer, 2003.

[51] Colin Percival. Stronger key derivation via sequential memory-hard functions. *Self-published*, pages 1–16, 2009.

[52] Orr Dunkelman and Dmitry Khovratovich. Iterative differentials, symmetries, and message modification in blake-256. In *ECRYPT2 Hash Workshop*, volume 2011. Citeseer, 2011.

[53] anonymous. Hashx11. http://cryptorials.io/glossary/x11/, 2014.

[54] Mohamed El-Hadedy, Martin Margala, Danilo Gligoroski, and Svein J Knapskog. Resource-efficient implementation of blue midnight wish-256 hash function on xilinx fpga platform. In *Information Assurance and Security (IAS), 2010 Sixth International Conference on*, pages 44–47. IEEE, 2010.

[55] Bernhard Jungk, Steffen Reith, and Jürgen Apfelbeck. On optimized fpga implementations of the sha-3 candidate groestl. *IACR Cryptology ePrint Archive*, 2009:206, 2009.

[56] Hongjun Wu. The hash function jh. *Submission to NIST (round 3)*, page 6, 2011.

[57] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. The keccak sha-3 submission. *Submission to NIST (Round 3)*, 6(7):16, 2011.

[58] Niels Ferguson, Stefan Lucks, Bruce Schneier, Doug Whiting, Mihir Bellare, Tadayoshi Kohno, Jon Callas, and Jesse Walker. The skein hash function family. *Submission to NIST (round 3)*, 7(7.5):3, 2010.

[59] Christophe De Canniere, Hisayoshi Sato, and Dai Watanabe. Hash function luffa: specification. *Submission to NIST (Round 2)*, 2009.

[60] Daniel J Bernstein. Cubehash specification (2. b. 1). *Submission to NIST*, 2008.

[61] Eli Biham and Orr Dunkelman. The shavite-3 hash function. *Submission to NIST (Round 2)*, page 113, 2009.

[62] Stefan Tillich, Martin Feldhofer, Mario Kirschbaum, Thomas Plos, Jörn-Marc Schmidt, and Alexander Szekely. High-speed hardware implementations of blake, blue midnight wish, cubehash, echo, fugue, gröstl, hamsi, jh, keccak, luffa, shabal, shavite-3, simd, and skein. *IACR Cryptology ePrint Archive*, 2009:510, 2009.

[63] Martin Schläffer. Subspace distinguisher for 5/8 rounds of the echo-256 hash function. In *International Workshop on Selected Areas in Cryptography*, pages 369–387. Springer, 2010.

[64] anonymous. Cryptonight. https://cryptonote.org/cns/cns008.txt, 2014.

[65] William J Luther. Cryptocurrencies, network effects, and switching costs. *Contemporary Economic Policy*, 2015.

[66] Mark Karpeles. Clarification of mt. gox compromised accounts and major bitcoin sell-off, 2011.

[67] anonymous. Failed cryptocurrencies. https://ltcflux.wordpress.com/2013/04/11/failed-cryptocurrencies-do-exist/, 2014.

[68] David Kuo Chuen LEE. The cryptocurrency revolution and its impact. *Self-published*, 2014.

[69] anonymous. Fail qubic. https://bitcointalk.org/index.php?topic=112676.0, 2014.