How Cryptography Benefits our Daily Life: An Overview on Blockchain System and Related Cryptographic Methods

Zerui Cheng

Institute for Interdisciplinary Information Sciences, Tsinghua University. (e-mail: cheng-zr19@mails.tsinghua.edu.cn)

Abstract: Based on several highly-cited papers (as listed in references), this is a survey on the prominent digital platform today: blockchain system as well as some related cryptographic methods applied in its protocol and consensus. To be more concrete, in the first half, I'll first give an introduction to the blockchain system, and then analyze the protocol of Bitcoin in particular. Also, I will briefly sketch other instances of blockchains and some progress in frontier at the end of this part. In the second half, I'll talk about the cryptographic methods applied in blockchain system, including SHA-256 algorithm used in incrementing a nonce value in a block and elliptic curves used in digital signatures. Last, I'll shed light on the current boundedness and limitations of blockchain, which suggests directions for further research in this topic.

Keywords: Blockchain, Bitcoin, Hash function, SHA-256, Digital signature, Elliptic curves

1. PREFACE

Recently, with the rapid development of the Internet and digital economy, we can observe significant and essential changes that happened to our daily life. Based on market capitalization, digital companies have occupied 7 positions among the largest companies all over the world till now [1], which indicates that the life on the Internet is the new trend of the historical process. Thus, in the new era with the theme of digital economy, protection of security and privacy over the Internet has become one of the major focal points of the field of computer science and technology. And many solutions have been put forward to solve the problem of privacy and security.

Among them, blockchain system [2] is sure to be one of the most prominent solutions. Blockchain system is a decentralized digital trust platform, where every user of the platform is equal (for example, the one-CPU-one-vote manner in Bitcoin) and there's no authority or supervisor at all. The privacy and security is based on cryptographic methods and computational hardness rather than humanities and trust in person, which is far more reliable. Blockchain system themes decentralization, anonymity, truly P2P mode and definite trust, and it came into prominence through the birth of Bitcoin, a cryptocurrency introduced by S.Nakamoto in 2008. [3]

Thus, as the project of the cryptography course, I investigate the blockchain system, a practical use of cryptography in our daily life. And this is a survey on blockchain system and related cryptographic methods applied in blockchain system. [4]

2. INTRODUCTION TO BLOCKCHAIN

2.1 Motivation and Basic Concepts

In a society, if we use an appropriate model to demonstrate the profits and losses, a reasonable assumption is that everybody is selfish and interest-driven, and thus none of the members in a society is unconditional trustful and reliable, resulting in the inherent weaknesses of trust-based models which is commonly adopted today (for example, the common currency in circulation now is based on our trust of the government), since they can be easily attacked by interest-driven malicious users, especially when the malicious users become the authorities or coordinators.Thus we need a more reliable and more robust mechanism to resolve the weaknesses for the whole community.

All above eventually motivates the idea and the birth of blockchain. And let's take a look at what blockchain is.

Blockchain, as a type of data structure, is a linked list that uses hash pointers instead of regular pointers. Here, a block is a data type that contains a particular header field, called the 'hash pointer' and some 'data' (for cryptocurrency, the data are usually transactions of coins). And the hash pointer field is simply the hash of another block, which we call its parent. A sequence of such blocks form a chain called blockchain.

[1] The data is from lecture notes of Prof.Pramod Viswanath in UIUC: *https://courses.grainger.illinois.edu/ece598pv/sp2021/#Lectures*

[2] In this article, if not specified, blockchain denotes public chain which is decentralized. Those centralized variants of blockchain such as consortium chains aren't considered in this essay.

[3] S.Nakamoto.2008.*Bitcoin: A Peer-to-Peer Electronic Cash System*

[4] Spoiler: Since this article serves as a survey, it will be more introductory rather than theoretical, and the main goal of this essay is to sketch an outline of current research in the subfield of blockchain. Thus most of the contents will be shown as direct results without concrete proof, especially those complicated but tedious ones. Readers can seek for the proof in the references if interested.

Furthermore. a blockchain system is much more complicated than blockchain as a data structure. Blockchain system is a digital trust system, where the trust is based on computational hardness and thus unbreakable. Since the blockchain data structure lies at the heart of such digital trust systems because it enjoys a good property of tamper-proof owing to its use of hash pointers, we use the same term to denote such a digital trust system.

The pure definition is pale and doesn't usually give a good intuition. Thus we furthermore look into the most popular blockchain system: Bitcoin to gain a deeper understanding of blockchain (Note that blockchain isn't only Bitcoin and cryptocurrency, but here we introduce Bitcoin as an appetizer since it's the most frequently mentioned and used instance of blockchain).

2.2 Introduction to Bitcoin

To elaborate the Bitcoin system from an original view, most contents in this subsection are based on the paper written by S.Nakamoto in 2008 [5]. However, as time goes, some contents of this paper are shown to be incorrect and Bitcoin system suffers from some attacks such as selfish mining [6], eclipse attack [7], block-withholding attack [8], stubborn mining [9] and so on, which is beyond Nakamoto's original expectation, the original genius idea still enjoys good reputation and is well worth spending time learning.

2.2.1 A Sketch of Bitcoin

Bitcoin system is an electronic payment system based on cryptographic proof instead of trust. Here, transactions are computationally impractical to reverse. Moreover, we don't need a trusted third party to supervise and validate all transactions any more. Bitcoin system is implemented by peer-to-peer distributed timestamp server, which generates computational proof of the chronological order of transactions. Also, Nakamoto claims that the system is secure as long as honest nodes collectively control more CPU power than any cooperating group of attacker nodes. (However, the attacks listed above proved this conjecture wrong.But good news is that, the threshold for satety isn't decreased too much. Take selfish mining as an example, assuming equal network condition, a malicious miner needs at least around 25.7% of overall computational power to gain positive profit.)

2.2.2 Transfer of a Coin

An electronic coin in the Bitcoin system is defined as a chain of digital signatures, which denotes the history of transaction of the coin. To transfer a coin, the sender digitally sign a hash of previous transaction and public key of next owner, then append it to the end the coin.Here, the signatures indicate chain of ownership and can be verified by the payee as well as the public (The concrete scheme is specified in Section 3.4 of the article).

However, the mechanism above can't prevent double-spending. To prevent it, a common solution is to add an intermediate central authority, however, then the system won't be decentralized any more, and we can hardly see its advantages compared to a bank. Then, Nakamoto put forward the following idea to prevent double-spending.

In all transactions of a single coin by its owner, only the earliest transaction counts and we don't care about later attempts to double-spend. To achieve this, transactions must be publicly announced. And participants should agree on a single history of the order in which they were received (the consensus will be shown in Section 2.2.3). Thus, the payee needs proof that at the time of each transaction, the majority of nodes agreed it was the first received. All these can be implemented by a timestamp server, where each timestamp includes the previous timestamp in its hash, forming a chain, with each additional timestamp reinforcing the ones before it.

2.2.3 PoW System (Proof-of-Work)

PoW (Proof-of-Work) system (where the currently popular word "mining" comes from) is widely used in blockchain, which results in one-CPU-one-vote manner of the whole community, making malicious behaviors nearly impossible if most users in the community are honest. And let's take a look at how it works.

The system is very simple. To guarantee the "one-CPU-one-vote" principle, we need each user of the system to provide a proof of how much computational power it has. And to generate the proof in such a decentralized system, the user should actually do some computation for the proof and the process of computation is called "mining". In Bitcoin system, miners are asked to scan for a value (called nonce) such that, when hashed by SHA256D algorithm, it begins with a number of 0-bits (adjustable,about 70-80 nowadays). Detailed process and how it preserves security and yields the "one-CPU-one-vote" manner is shown in Section 3.3.

Once a miner finds such a nonce value, he is granted the right to add a block after the public chain. Then he adds a block to the chain, where the data includes all transactions happened in this time interval in the block as well as the nonce value he finds. Other miners can verify whether the nonce value is a valid one by executing a single hash and admit it if valid.

To compensate for increasing hardware speed and varying interest in running nodes over time, PoW difficulty is determined by a moving average targeting an average number of blocks per hour. In practice, the system modifies the difficulty every 2016 blocks and control the speed to be around 10 minutes per block.

2.2.4 Longest Chain Rule (Nakamoto Consensus)

From the previous arguments, when a node finds a proof-of-work, it broadcasts the block to all nodes, and nodes accept the block only if all transactions in it are

[5] S.Nakamoto.2008.Bitcoin: A Peer-to-Peer Electronic Cash System

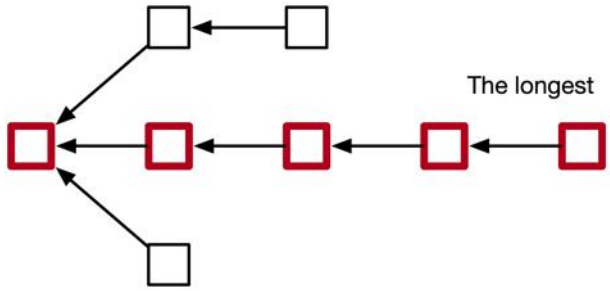
[6] Eyal, Ittay, and Emin Gün Sirer. "Majority is not enough: Bitcoin mining is vulnerable." International conference on financial cryptography and data security. Springer, Berlin, Heidelberg, 2014.

[7] Heilman, Ethan, et al. "Eclipse attacks on bitcoin's peer-to-peer network." 24th USENIX Security Symposium (USENIX Security 15). 2015.

[8] Eyal, Ittay. "The miner's dilemma." 2015 IEEE Symposium on Security and Privacy. IEEE, 2015.

[9] Nayak, Kartik, et al. "Stubborn mining: Generalizing selfish mining and combining with an eclipse attack." 2016 IEEE European Symposium on Security and Privacy (EuroS&P). IEEE, 2016.

valid and not already spent. In the ideal case, the public chain will be a chain, where every block is confirmed at last. However, in practice, since in each round of trying to find a PoW, everyone is equal position and there exists network delay, there may be branches of chains as shown in the following figure.



Then the famous Nakomoto Consensus (also called the longest chain rule) says that, at any time, for a miner, he always mines on the end of the longest chain from his view of the whole system (Note that the view from a miner may be different from the exact state of the state of the system, due to network delay). The scheme guarantees that the consensus on transactions are based on majority of the community, and thus a small number of hackers won't affect the system. And in this way, only transactions packed in the blocks that are on the longest chain are confirmed by the community, while the transactions in the orphaned blocks should be re-broadcast to miners to be packed on the longest chain.

2.2.5 Other Aspects in Bitcoin

Here we briefly mention some other aspects in Bitcoin, although they're less related to the main topic.

To help miners in the community stay honest, the incentive mechanism of Bitcoin protocol is as follows. First, the first transaction in a block is a special one starting a new coin owned by the creator of the block. It's an incentive for nodes to support the network, and is also a way to initially distribute coins into circulation without central authority. Second, the incentive can also be funded with transaction fees. If the output value of a transaction is less than its input value, the difference is a transaction fee that is added to the incentive value of the block. Moreover, once a predetermined number of coins have entered circulation (21 million in Bitcoin), the incentive only contains transaction fees and the system is completely inflation free.

For privacy, although all transactions should be public, privacy can still be maintained by keeping public keys anonymous. However, the public can see that someone is sending an amount to someone else. As time goes by, if the key pair isn't changed, the amount of money of each user can be figured out by the public (although the public doesn't know who exactly it attributes to, but it's dangerous for those with large amount of coins, since the number of such users is limited). Thus a new key pair should be used for each transaction to keep privacy.

2.3 Other Instances of Blockchain

Apart from Bitcoin, there also exists many variants of Blockchain system. For example, the second-largest cryptocurrency, Ethereum [10], introduces the concept of uncle block as a generalization in contrast to the concept of parent block in Bitcoin as well as the concept of gas limit and gas fee, which decreases the latency, increases the throughput, increases the motivation of miners and is regarded as the second generation of blockchain. Also, Hyperledger [11] is a new kind of blockchain which doesn't rely on anonymous miners for verifying transactions, and thus doesn't have corresponding cryptocurrency as a way of incentive, but users should get permission to enter the system. It gets rid of the inherent weakness of cryptocurrency that the market will make the price unstable and thus affect the technical development of blockchain (as the users usually want to gamble on the price instead of enjoying the advantages of decentralization and trust). Hyperledger is now widely used in enterprise-grade blockchain deployments and building distributed system. Moreover, Conflux [12] is also a prominent blockchain system put forward by members of IIIS, which generates a topological order of blocks so that none of the blocks will be orphaned any more and thus rejects selfish mining or similar attacks from the essence. Also, it makes use of the Link-Cut Tree data structure which is familiar to those who learnt OI to speed up the order generation process, which results in prominent throughput. Moreover, it uses GHAST [13] protocol to increase confirmation speed, which results in small latency that dominates Bitcoin in the order of magnitude and has more potential in the future.

The instances mentioned above all employs PoW system mentioned in Section 2.2.3, which causes severe environmental problem since the computation process is simply exchanging power for nothing but a nonce value (although it helps make the community stable). Thus, a new idea of PoS (Proof-of-Stake) system [14] comes into our sight and its idea is implemented on PPCoin initially. Instead of the "one-CPU-one-vote" PoW system which is much too energy-consuming and environmentally-unfriendly, it decides users' significance of a vote by the amount of coins they have, which results in a "one-coin-one-vote" mode. The miners only need to submit a proof of their coins to get the right to append a block after the chain without solving complicated, tedious and meaningless cryptographic puzzle. Also, since only those

[10] Buterin, Vitalik. "A next-generation smart contract and decentralized application platform." white paper 3.37 (2014).

[11] Androulaki, Elli, et al. "Hyperledger fabric: a distributed operating system for permissioned blockchains." Proceedings of the thirteenth EuroSys conference. 2018.

[12] Li, Chenxing, et al. "A decentralized blockchain with high throughput and fast confirmation." 2020 USENIX Annual Technical Conference (USENIXATC 20). 2020.

[13] Li, Chenxing, Fan Long, and Guang Yang. "GHAST: Breaking confirmation delay barrier in nakamoto consensus via adaptive weighted blocks." arXiv preprint arXiv:2006.01072 (2020).

[14] King, Sunny, and Scott Nadal. "Ppcoin: Peer-to-peer crypto-currency with proof-of-stake." self-published paper, August 19 (2012): 1.

who have enough coins can operate on the chain, they don't have motivation to break the system since any harm to the system will only lead to devaluation of their coins. However, it suffers from the nothing-at-stake attack since the hacker with no stake has nothing to lose and is thus not a perfect solution. One compensation for this is DPoS [15] system where users with no stake don't have access to the system, however, in this case, a central authority is needed and thus the system isn't decentralized any more. Nevertheless, PoS system is still in the frontier of research in blockchain, and is expected to substitute PoW system to be a popular consensus system applied in the next-generation blockchain.

3. HOW CRYPTO APPLIES IN BLOCKCHAIN

3.1 Specification of some Definitions

Before stepping into the concrete contents on how cryptography applies in blockchain, we first recapture some definitions that may be useful in the latter part.

Definition 3.1. (Hash Function)

A hash function is a function H that converts a binary string x of arbitrary length to a binary string $H(x)$ of fixed length with the following two properties:

- *Easy to compute*: $\exists$ n.u.p.p.t A that computes $H(x)$ for any input x with success probability 1;
- *Collision-reducible* : Suppose the input space of x is X and the output space of $H(x)$ is Y, then $H(x)$ should distribute uniformly on Y when we go through all possible input $x \in X$.

Note that the definition above doesn't require the property of "Hard to Invert" or "Collision-Resistant" which we emphasize in class. It has been widely used in Hash Tables but doesn't make much sense in cryptography. Thus, such a hash function isn't what we need and we can strengthen the definition as follows.

Definition 3.2. (Cryptogragphic Hash Function, CHF)

A cryptographic hash function is a hash function $H : \{0,1\}^m \to \{0,1\}^n$ with the following property in addition:

- *Collision-resistant* : Any n.u.p.p.t Adv can't come up with $x \neq x'$ satisfying $H(x) = H(x')$ with non-negligible probability. (i.e. Given $x \in X$, there's no exponential speed-up to find $x' \neq x$ satisfying $H(x) = H(x')$ in contrast to exhaustive search).

Note that, as mentioned in class, "Collision-resistant" property implies the function is at least one-way. Also, particularly, in the following part, a feasible hash function that can be applied in blockchain should satisfy the following property of "puzzle-friendly" in addition:

Definition 3.3. (Puzzle-friendly CHF, PCHF)

A puzzle-friendly cryptographic hash function is a CHF $H : \{0,1\}^m \to \{0,1\}^n$ with the following property in addition:

- *Puzzle-friendly* : Any n.u.p.p.t Adv can't come up with x satisfying $H(x) \in Y \subset \{0,1\}^n$ where $\frac{|Y|}{2^n} \leq \epsilon(n)$ for some negligible ϵ with non-negligible probability. (i.e. It's just a generalization of the "one-way" property. Note that here $|Y|$ doesn't necessarily be polynomial size, for example, $|Y| = 2^{\frac{n}{2}}$).

Above are the useful definitions involved in the following. Moreover, let's review the concept of digital signatures, which is an analog of handwritten signatures in the context of cryptography. And they're used when sending transactions where the receiver will request a digital signature from the sender to verify the fidelity of the transaction.

Definition 3.4. (Digital Signature)

$(Gen, Sign, Ver)$ is a digital signature scheme over the message space $\{M_n\}_n$ if

- $Gen(1^n)$ is a p.p.t. which on input n outputs a public key p_k and a secret key s_k, i.e. $p_k, s_k \leftarrow Gen(1^n)$
- $Sign$ is a p.p.t. which on input a secret key s_k and message m outputs a signature σ, i.e. $\sigma \leftarrow Sign_{s_k}(m)$
- Ver is a deterministic p.p.t. algorithm which on input a public key p_k, a message m and a signature σ returns either "accept" or "reject". And it satisfies the following property: For all $m \in M$,

$$Pr[p_k, s_k \leftarrow Gen(1^n) : Ver_{p_k}(m, Sign_{s_k}(m)) = accept] = 1$$

Every time for a transaction, the sender generates the public key and the secret key and publishes the public key. Afterwards, the sender runs the $Sign$ procedure and appends the signature after the coin to send it to the receiver. Upon receiving it, the receiver as well as the public can run Ver procedure to verify the authentication of the coin received.

However, for blockchain, only those properties aren't enough, since a malicious user can fabricate a signature to confuse the receiver, thus we need to add the property of "unforgeable" for a feasible digital signature.

Definition 3.5. (Unforgeable Digital Signature)

A digital signature scheme $(Gen, Sign, Ver)$ is unforgeable over the message space $\{M_n\}_n$ if it satisfies

- (unforgeable) For any n.u,p.p.t Adv, there exists negligible $\epsilon(\cdot)$ such that

$$Pr[p_k, s_k \leftarrow Gen(1^n) : Ver_{p_k}(m, Adv(m)) = accept] \leq \epsilon(n)$$

[15] Fan, Xinxin, and Qi Chai. "Roll-DPoS: a randomized delegated proof of stake scheme for scalable blockchain-based internet of things systems." Proceedings of the 15th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services. 2018.

We can note that the definition of "unforgeable" is similar to "security" that we mentioned in class. The unforgeable digital signatures can be applied in the protocol of Bitcoin, as will be mentioned in section 3.3. In short, in an unforgeable digital signature scheme, the signer generates secret key and public key and publishes the public key, others can use the public key p_k to verify that the signer actually has the secret key with probability 1, and the signature is unforgeable without knowing the secret key.(i.e. The probability to fabricate successfully within polynomial time is negligible).

Then, we look into some concrete methods applied in blockchain, as shown in the following.

3.2 Merkle-Damgard Construction

Cryptographic methods are used in various kinds of blockchains. As mentioned before, we need to create a hard cryptographic puzzle to guarantee the "one-CPU-one-vote" scheme. Thus, we need a cryptographic hash function (Definition 2.2), and following is a popular construction applied in blockchains.

If we already have a PCHF from $\{0,1\}^m$ to $\{0,1\}^n$ where $m > n$, we may want to generalize it to one from $\{0,1\}^t$ to $\{0,1\}^n$ for arbitrarily larger $t > m$ while reserving its property as PCHF. And a general principle used in the construction of many cryptographic hash functions is the Merkle-Damgard construction as follows. For a message $x \in \{0,1\}^t$ for arbitrary long t, we first break m into portions and call them $x_1, x_2, ...$, where x_1 is of length m and others are of length $m - n$ (append 0 on the last if the last portion isn't long enough). Then we can apply the PCHF $F : \{0,1\}^m \to \{0,1\}^n$ recursively, for the first iteration, the input string is $u_1 = x_1$ and we get $F(u_1) = y_1 \in \{0,1\}^n$. Then for the i^{th} iteration where $i \geq 2$, the input string is $u_i = y_{i-1} \circ x_i \in \{0,1\}^{n+(m-n)} = \{0,1\}^m$, and the output string is $F(u_i) = y_i \in \{0,1\}^n$. Finally, the output string of the generalized hash function is y_c if we suppose that the input string is divided into c portions. We can prove that the generalized hash function still holds the property of collision-resistant by contradiction, and thus the Merkle-Damgard construction also results in a cryptographic hash function from $\{0,1\}^t$ to $\{0,1\}^n$.

And this generalization is widely used in creating a cryptographic puzzle in blockchain system.

3.3 SHA-256 Algorithm Applied in Bitcoin

First we clarify the name of SHA-256 [16] . Here, SHA stands for "secure hash algorithm" and 256 denotes the bits of the image of the hash function. And SHA-256 algorithm is put forward by NIST [17] in 2001, which is used in Bitcoin as the cryptographic puzzle for miners to solve to increment the nonce of a block.

The original version of SHA-256 algorithm hashes a 768-bit string to a 256-bit string, which is considered to be a secure PCHF [18] . Its scheme involves complicated bitwise operation, and we will omit it here since it's tedious and doesn't have something interesting to say. [19] Based on the original version of SHA-256 which compresses a 768-bit string to 256-bit, we can generalize it using Merkle-Damgard construction shown above to compress a string of any length to 256 bits. Moreover, in most theoretical analysis, we usually don't care the implementation of SHA-256, but simply regard it as PCHF for derivation.

In the Bitcoin protocol, for a miner to be granted the right of adding a new block to the public chain, the miner should find a nonce value, which satisfies that, the SHA-256 hash result of the previous hash concatenated by the nonce value has multiple consecutive 0s in the beginning (The exact number of 0s required is adjustable depending on the total computational power of the whole system, in order to guarantee constant speed of generation of new blocks. And the number is around 70-80 today while it's only about 60 at the brith of Bitcoin). Since SHA-256 is considered to be PCHF, any miner can only use exhaustive search to find such value. Thus, each round of competing to generate a new block is essentially a Poissonian process, and at each round, the probability that a particular miner first comes up with the nonce value is proportional to its computational power. Thus the expected proportion of the miner's blocks on the whole chain is determined by its computational power, and the true proportion will converge to the expectation in the long term from the central limit theorem. As a result, if most of the community is honest, a malicious miner can hardly hurt the system since the authenticated blocks produced by him is very limited, and his malicious behavior can be easily compensated by the following block (i.e. A malicious miner may choose to ignore some transactions that are already signed, however, the next honest miner can put them into his block to verify the transactions), which won't hurt the system at all, thus proving the security of robustness of the whole system, which is guaranteed by the PCHF property of SHA-256 Algorithm.

3.4 Elliptic Curves for Digital Signatures in Bitcoin

In the scheme of Bitcoin, we use ECDSA [20] (Elliptic Curves Elliptic Curve Digital Signature Algorithms) for signing a transaction, and ECDSA is proved to be cryptographically secure even against an adaptive adversary. And Bitcoin applies the algorithm of SECP256K1 in the family of ECDSA.

Then, let's shed light on how ECDSA works. And following gives an intuition on its working scheme. [21]

[16] Gilbert, Henri, and Helena Handschuh. "Security analysis of SHA-256 and sisters." International workshop on selected areas in cryptography. Springer, Berlin, Heidelberg, 2003.

[17] NIST:"National Institute of Standards and Technology".

[18] There's no provable security for it until today, but it's considered to be secure since there's also no valid attack to it.

[19] For those who are interested in the proof, you can refer to https://en.wikipedia.org/wiki/SHA-256 for its concrete scheme.

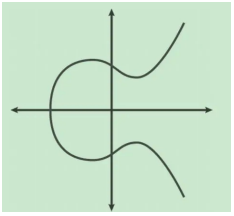
[20] Johnson, Don, Alfred Menezes, and Scott Vanstone. "The elliptic curve digital signature algorithm (ECDSA)." International journal of information security 1.1 (2001): 36-63.

[21] The proof why ECDSA is cryptographically secure involves hard and complicated maths knowledge, thus we omit it since our article serves as a survey. Readers who are interested can refer to the paper above.

As shown in its name, we first need an elliptic curve as the basis of our algorithm. And the elliptic curve is in the form:

$$y^2 \equiv x^3 + ax + b \pmod{p}$$

And the curve is plotted in the following figure.



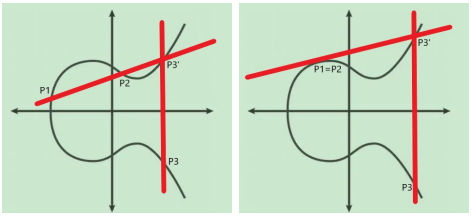
Then, some involved maths derivation shows us that, for all points (x, y) on the curves such that $x, y \in F_p$, they essentially form a field. And we use $E(F_p)$ to denote it, which is called the elliptic curve field specified by the function above.

For the field $E(F_p)$, the addition is defined as follows:

Suppose $P_1, P_2 \in E(F_p)$ where $P_1 = (x_1, y_1), P_2 = (x_2, y_2)$, then $P_1 + P_2 = P_3 = (x_3, y_3)$, where

$$x_3 \equiv \lambda^2 - x_1 - x_2 \pmod{p}$$
$$y_3 \equiv \lambda(x_1 - x_3) - y_1 \pmod{p}$$

Here $\lambda \equiv (x_2 - x_1)^{-1}(y_2 - y_1) \pmod{p}$ if $P_1 \neq P_2$ and $\lambda \equiv (2y_1)^{-1}(3x_1^2 + a) \pmod{p}$ if $P_1 = P_2$, where x^{-1} denotes the inverse of x modulo p. The intuition is that, if $P_1 \neq P_2$, draw a line connecting them and it will intersect with the curve on another point P_3', and draw a line parallel to y-axis which passes through P_3', then the intersection point is P_3. When $P_1 = P_2$, the line will be the tangential line going through P_1 other than a secant line, as shown in the following. (The first denotes the case $P_1 \neq P_2$, while the second demonstrates $P_1 = P_2$)



For multiplication with a constant number k, it is simply recursively applying the addition operation for k times, which is consistent with the normal multiplication we do in algebra. In fact, the mathematical proof of the security of elliptic curve is that, given $B = k \times A$ and A, we can't restore k, since the minus and division on elliptic curves doesn't exist. The trapdoor function property of multiplication of elliptic curves guarantees the security.

Then we select N points on the curve and choose one of the points to be the origin point G, here N is the size of the message space and we can make a projection from all possible messages to the range $[0, N-1]$. The selection of the points will depend on the cofactor $h = \frac{\#(E(F_p))}{N}$ (i.e. density of selected points), where $E(F_p)$ is the elliptic curve field specified by the function above, which is composed of points on the curve. Then the 6-tuple (a, b, p, N, G, h) specifies an elliptic curve, which is also known to the public. Note that p, N should both be prime (we can always find appropriate p and N according to the distribution of primes). In particular, for Bitcoin, we choose $a = 0$, $b = 7$ and p, N are both large primes around 2^{256},laying the foundation of the algorithm.

Then we will do some magic on the elliptic curve in ECDSA. For the Gen process, the secret s_k is a purely random number. And we can compute $Q = s_k \times G$ as another point on the elliptic curve and take Q as the public key p_k. Then, for the $Sign$ process, first generate a random number k and compute the point $P = k \times G$ on the curve, then suppose z is the hash value of the message M to sign, P_x is the x-coordinate of the point P , then it computes the following value

$$S \equiv k^{-1}(z + s_k \times P_x) \pmod{p}$$

Then it outputs $\{P_x, S\}$ as the signature σ. Finally, for the verifier $Ver_Q(z, \{P_x, S\})$, it can restore the point P computed by Gen in the following manner:

$$P = S^{-1} \times z \times G + S^{-1} \times p_k \times Q$$

Then it compares P_x in σ with the x-coordinate of the computed point P. If they're equal, then the signature is valid and $Ver_Q(z, \{P_x, S\})$ returns accept. Otherwise, the signature is an invalid one and $Ver_Q(z, \{P_x, S\})$ rejects.

Above is the protocol for signing a digital signature in Bitcoin system. In practice, every user of the Bitcoin system holds a pair of (s_k, p_k) where s_k is kept private and p_k is broadcast to all users, what's more, p_k is the only certificate of identity in the community. To sign a transaction, the sender signs the previous hash and the receiver's public key as the message in the digital signature and append it on the end of the coin. Then, the miners can verify the fidelity of the transaction by running the Ver process. Once the transaction is verified, it goes to the waiting list to be packed in a block. After the block is appended to the chain and gets confirmed by majority of the community (A usual criterion is for Bitcoin that, 6 blocks generated behind the block are all on the longest chain), the transaction eventually comes to a success.

4. CURRENT LIMITATIONS OF BLOCKCHAIN AND DIRECTIONS FOR FURTHER RESEARCH

However, as a newly-born subfield, blockchain is now still in its primary stage and has some issues remaining to be solved. These are bad news in a way, but also good news in another sense, since it also implies that there's a lot of things to do in this area and the issues suggest directions for further research. And let's take a look at some of them.

- Social problems: There're three major social problems that the government may concern about, and that's also the reason why our country has published more restrictions on blockchain recently.
  (1) The corresponding cryptocurrency of blockchain now has roller-coaster-like curve in its value compared to legal currency.Thus it attracts some gamblers to put a lot of money into the system. Although it helps the blockchain system gain more attention and more popularity in a sense, the gamblers are potentially unstable factors of

the society since they may lose everything in cryptocurrency and thus do something harmful to the whole society.

(2) The blockchain system now lacks effective supervision. Since one can trade the illegal money for cryptocurrency and use cryptocurrency to trade, the blockchain system may become the paradise for illegal guys. Moreover, the trade of guns and drugs can also be conducted on blockchain today. Thus supervision on blockchain is necessary, but how to carry out over such a decentralized system remains to be a problem.

(3) What's more, since cryptocurrency themes decentralization, it's a threat to the government and the authorities. Thus, how to make blockchain system and government co-exist in harmony, is also a subject for the policy makers in the government as well as users of blockchain.

Usually solving these kinds of problems isn't in the sight of researchers, especially those who do purely theoretical research. However, to maintain a relatively loose atmosphere for research, these problems aren't only for the government to solve, for the researchers as well as the public, it's always beneficial to think about the resolution to the problems. And only in this way can blockchain get rid of the so-called "game of gamblers" and eventually become a new trust system that benefits the daily life of everyone.

Nowadays, for the third aspect I mention above, a new kind of blockchain system, called consortium chain, serves as a concession to the government and has gained more and more popularity. This kind of blockchain system is no more decentralized and needs an authority to give users access to enter. It's often applied in the case where several parties would like to share a distributed system to do something together. And the digital trust property of blockchain will discard the worry that any of the parties would do something malicious. Moreover, the most important point is that, it can be supervised since the government only needs to keep an eye on the authority to grant access. The consortium chain is now a newly-born area, the problems about how to generalize it to more parties and how to guarantee the digital trust with a centralized authority are interesting to work on and a direction for further research.

- Environmental problems: As mentioned previously, PoW system consumes too much energy and power, and thus violates the purpose of sustainable development. Thus one direction for research is to put forward new consensus mechanism to substitute it. PoS and DPoS (as mentioned in Section 2.3) are both good attempts, and the completion and addition on the basis of PoS system now draws the interest of some related researchers, and how to prevent nothing-at-stake attack in PoS is a research interest today. If the problem of nothing-at-stake attack is tackled, then PoS can be widely used in blockchain, which indicates a new generation and a new era of blockchain system.
- Technical problems: Moreover, there're still many technical problems with regard to how to improve the performance of blockchain, such as limited throughput and high latency. In any of the blockchain system nowadays, the problems of throughput and latency are common. The maximum capacity of transactions per unit time is limited, and the confirmation time for current blockchain is also too long. They both restrict blockchain system from getting more popular and being applied worldwide since it can't hold the amount of transactions as WeChat Pay and Alipay today, and users can't tolerate a long confirmation time, either. Now, in Conflux, Link-Cut Tree and GHAST are applied to improve these aspects, although the effect is prominent, it's far from the expected stage to be commonly used. Thus it also suggests a direction for further research.

In conlusion, once these limitations are tackled, I'm confident that blockchain will enjoy more popularity all over the world, and can thus benefit the daily life of everyone, making the society work in a more robust mode, giving every single person a more secure and comfortable environment to live and work in.

5. REFERENCES

[1] S.Nakamoto.2008.Bitcoin: A Peer-to-Peer Electronic Cash System.

[2] Coron, Jean-Sébastien, et al. "Merkle-Damgård revisited: How to construct a hash function." Annual International Cryptology Conference. Springer, Berlin, Heidelberg, 2005.

[3] Gilbert, Henri, and Helena Handschuh. "Security analysis of SHA-256 and sisters." International workshop on selected areas in cryptography. Springer, Berlin, Heidelberg, 2003.

[4] Johnson, Don, Alfred Menezes, and Scott Vanstone. "The elliptic curve digital signature algorithm (ECDSA)." International journal of information security 1.1 (2001): 36-63.

[5] Buterin,Vitalik."A next-generation smart contract and decentralized application platform." white paper 3.37 (2014).

[6] Androulaki,Elli,et al. "Hyperledger fabric: a distributed operatingsystem for permissioned blockchains." Proceedings of the thirteenthEuroSys conference. 2018.

[7] Li,Chenxing,et al."A decentralized blockchain with high throughput and fast confirmation." 2020 USENIX Annual TechnicalConference (USENIXATC 20). 2020.

[8] Li,Chenxing, Fan Long, and Guang Yang. "GHAST: Breaking confirmation delay barrier in nakamoto consensus via adaptive weighted blocks." arXiv preprint arXiv:2006.01072 (2020).

[9] King, Sunny, and Scott Nadal. "Ppcoin: Peer-to-peer crypto-currency with proof-of-stake." self-published paper, August 19 (2012): 1.

[10] Fan, Xinxin, and Qi Chai. "Roll-DPoS: a randomized delegated proof of stake scheme for scalable blockchain-based internet of things systems." Proceedings of the 15th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services. 2018.

[11] Eyal, Ittay, and Emin Gün Sirer. "Majority is not enough: Bitcoin mining is vulnerable." International conference on financial cryptography and data security. Springer, Berlin, Heidelberg, 2014,

[12] Heilman, Ethan, et al. "Eclipse attacks on bitcoin's peer-to-peer network." 24th USENIX Security Symposium (USENIX Security 15). 2015.

[13] Eyal, Ittay. "The miner's dilemma." 2015 IEEE Symposium on Security and Privacy. IEEE, 2015.

[14] Nayak, Kartik, et al. "Stubborn mining: Generalizing selfish mining and combining with an eclipse attack." 2016 IEEE European Symposium on Security and Privacy (EuroS&P). IEEE, 2016.

[15] Lecture notes of ECE 598CV in UIUC: Principle of Blockchains, instructed by Prof.Pramod Viswanath:
https://courses.grainger.illinois.edu/ece598pv/sp2021/#Lectures